

VAASAN YLIOPISTO
TEKNILLINEN TIEDEKUNTA
TIETOTEKNIIKAN LAITOS

Jyrki Kankaanpää
**AIKASARJAN MALLINTAMINEN MULTI-LAYER-
PERCEPTRON -NEUROVERKOLLA**

KTM, tietotekniikka
Pro Gradu – tutkielma

VAASA 2007

SISÄLLYSLUETTELO

Termit ja lyhenteet.....	4
TIIVISTELMÄ:.....	6
ABSTRACT:	7
1. JOHDANTO.....	8
1.1. Työn tavoite ja raja.....	10
2. TAUSTATEORIAM.....	12
2.1 Aikasarjat.....	12
2.1.1 Näytteistys ja summaaminen.....	13
2.1.2 Aikasarjojen ominaisuuksia.....	15
2.1.3 Ennustaminen ja aikahorisontit	18
2.1.4 Aikasarjojen mallinnusmenetelmiä	18
2.1.5 Aikasarjadatan esikäsittely	21
2.2 Neuroverkot.....	21
2.2.1 Neuroverkkojen esikuva.....	23
2.2.2 Neuroverkon rakenne	26
2.2.4 Neuroverkon opettaminen	29
2.2.5 Neuroverkon rakenteen suunnitteleminen	35
2.2.6 Datan esi- ja jälkikäsittely	36
2.3 Aikasarjamallin ennustuskäytön ja ennustevirheen mittaaminen.....	40
3. AIKASARJAN MALLINNUS JA ENNUSTEIDEN MUODOSTAMINEN	42
3.1 Visuaalinen tarkastelu ja esikäsittely.....	42
3.2 Neuroverkon konfigurointi.....	48
3.3 Neuroverkon opettaminen opetus-datalla.....	49
3.4 Ennusteiden muodostaminen.....	51
4 ANALYYSI.....	53
4.1 Ennusteen tarkkuus ja aikasarjamallin ennustuskäytö.....	53
5. YHTEENVETO	57
LÄHDELUETTELO	58
LIITE 1	59
Excel-sovelluksen VBA- koodi.....	59
LIITE 2	67
Työssä käytetty "raaka" aikasarjadata	67
LIITE 3	69
Neuroverkoissa käytetty "putsattu" aikasarja	69
LIITE 4	74
Koe-ennustukset	74
LIITE 5	76
Valitulla neuroverkolla tuotetut ennusteet.....	76

Termit ja lyhenteet

AI	Keinoäly (artificial intelligence).
Aikasarja	Peräkkäisinä ajan hetkinä mitattu muuttujan arvoista muodostuva data.
Aktivaatiofunktio	Neuroniin liitetty funktio, joka määrittää neuronin vasteen summatulle painotetulle syötteelle.
Back-propagatio	MLP neuroverkon opetusalgoritmi.
Bias	Neuroneiden summattuun syötteeseen lisättävä vakio.
Binääri sigmoidifunktio	Epälineaarisiin neuroverkkoihin, aktivaatiofunktiona käytetty derivoituva s-käyrän tuottava funktio, joka voi saada arvoja väliltä $[0,1]$.
Diskreetti aikasarja	Aikasarja, jossa muuttuja voi saada vain tiettyjä arvoja tiettyinä ajan hetkinä.
Jatkuva aikasarja	Aikasarja, jossa muuttuja voi saada äärettömästi arvoja.
MA	Liukuvan keskiarvon aikasarjamalli (moving average).
MAPE	Keskiarvoinen absoluuttinen suhteellinen virhe (mean absolute percentage error).
MAD	Keskiarvoinen absoluuttinen poikkeama (mean absolute deviation).
MLP	Multi layer perceptron eli monitasoneuroverkko.
MSE	Keskiarvoinen neliövirhe (mean squared error).
Neuroni	Neuroverkon keinotekoinen aivosolu.
Outliner	Aikasarjassa muista arvoista poikkeava arvo.
Painokerroin	Särmään liittyvä luku, jolla syötettä painotetaan.
Perceptron	Ohjatun oppimisen neuroverkkomalli.
Piiloneuroni	Neuroni joka sijaitsee ulostulon neuroneiden ja syöteneuroneiden välissä.

Sigmoidifunktio	Epälineaarisiin neuroverkkoihin, aktivaatiofunktiona käytetty derivoituva s-käyrän tuottava funktio, joka voi saada arvoja väliltä $[-1,1]$.
Särmä	Kytkee neuronit toisiinsa.
SOM	Self Organizing Map eli itse organisoituva kartta.
Syöte	Neuroverkolle annettu data.
Syötoneuroni	Neuroni joihin liitetään syötedata.
Vaste	Neuronin antama tulos.
Vastoneuroni	Neuroverkon antama vaste syötedatalle.
VBA	Ohjelmointikieli (visual basic for applications).

VAASAN YLIOPISTO**Teknillinen tiedekunta****Tekijä:**

Jyrki Kankaanpää

Tutkielman nimi:

Aikasarjan mallintaminen Multi-Layer-Perceptron -neuroverkolla

Ohjaajan nimi:

Merja Wanne

Tutkinto:

Kauppatieteiden maisteri

Laitos:

Tietotekniikan laitos

Oppiaine:

Tietotekniikka

Opintojen aloitusvuosi:

2004

Pro gradun valmistumisvuosi:

2007

Sivumäärä: 76

TIIVISTELMÄ:

Tässä työssä tutkitaan miten Multi-Layer-Perceptron -neuroverkolla mallinnetaan aikasarjoja ja miten luotua mallia voidaan käyttää tuottamaan ennusteita mallinnetusta aikasarjasta. Lisäksi konstruoidaan neuroverkkosovellus, jonka avulla luodaan ennusteita aikasarjasta ja tutkitaan ennusteiden luotettavuus.

Aluksi tässä työssä tutustutaan aikasarjoihin ja neuroverkkoihin. Aikasarjoista selvitetään miten ne määritellään kuuluvaksi joko jatkuva- tai diskreettiaikaisiin aikasarjoihin. Tutkitaan myös, mistä ja miten voidaan päätellä aikasarjan sisältävän informaatiota itsestään, sekä miten ne sen perusteella jaetaan stokastisiin ja deterministisiin aikasarjoihin. Pohditaan myös sitä, millaisissa prosesseissa aikasarjoja syntyy ja millaisia ominaisuuksia niillä on.

Neuroverkkoja tutkivassa osuudessa tutustutaan ensin lyhyesti niiden esikuvaan, aivoihin, jonka jälkeen tutustutaan keinotekoisen neuronin malliin ja back-propagation-algoritmiin. Varsinaisten ennusteiden muodostamista varten luodaan edellisten teorioiden pohjalta VBA-Excel -sovellus joka tuottaa neuroverkon. Neuroverkkoja luodaan kaksi ja niitä testataan opettamalla niitä eri opetuskierrosmäärillä ja parhaan ennustetuloksen antava verkko valitaan lopulliseksi aikasarjan malliksi.

Lopuksi tutkitaan neuroverkon luotettavuus ennusteen tuottajana. Tässä työssä tuotetun aikasarjan neuroverkkomalli toimi suhteellisen hyvin. Neuroverkkojen käyttö kuitenkin käytännössä saattaa tuottaa hankaluutta, niiden vaatiman yksittäisen nimikkeen aikasarjan opettamisen ja ennusteiden luomisen suhteen, mikäli nimikkeiden määrä on suuri.

AVAINSANAT: Aikasarja, neuroverkko, tekoäly, ennuste, alkukäsittely, loppukäsittely

UNIVERSITY OF VAASA
Faculty of technology
Author:

Jyrki Kankaanpää

Topic of the thesis:

Time series modelling by Multi-Layer-Perceptron neural network

Supervisor:

Merja Wanne

Degree:

Master of Science in Economics and Business Administration

Department:

Department of Computer science

Major of subject:

Computer science

Year of the entering the university:

2004

Year of the completing the thesis:

2007

Pages: 76

ABSTRACT:

How Multi-Layer-Perceptron neural network can model time series and how good that model is to produce forecast for the time series, is the question of this study. A neural network application is constructed and with that application forecasts are created.

In the beginning of this study, different time series and neural networks are introduced. The time series are defined to the stochastic or to the deterministic series. It is also studied, do the time series include information of themselves and what kind of properties time series do have.

Because the brains are the model of artificial neural network, brains are introduced in the first part of this study, which studies neural networks. After that, the artificial neuron, multi layer perception theory and back-propagation-teaching-algorithm are introduced and used to create a neural network, which is then needed to forecast the time series.

To create the actual forecast, a neural network based Excel-application is created. Several neural networks are created and studied, the network which gives the best forecast is then selected to be the final model for the time series.

Finally, the reliability to create a forecast, of the neural network, is studied and evaluated. The neural network model produced in this study, proved to be relatively reliable according to the forecasts. In practice, the use of neural network may be difficult, especially if there are many items, because every item must be taught individually, which means that every item needs a network of its own.

KEYWORDS: Time series, neural network, forecast, pre-processing, post processing

1. JOHDANTO

Tulevien tapahtumien ennakoiminen on yksi ihmisen luonteenomaisimpia piirteitä. Mikä olisikaan tuottoisampaa kuin pörssikurssien tietäminen ennakolta, toisaalta tuskin kukaan haluaa tietää tulevaisuudesta kaiken (Pynnönen 1999: 33). Voisi kuvitella ennalta tietämisen aiheuttavan suuresti ahdistusta jos tietäisi esim. läheisen kuoleman täsmällisen hetken ja kuolinsyyn.

Ennustusmenetelmiä on useita, jotkut tieteellisesti hyväksyttävä, toiset taas eivät. Ennustusvälineiksi on kelvannut mm. eläinten sisäelimet, kahvinporot, tähdet, sammakot (Pynnönen 1999: 33). Nykyisin joillain tieteenaloilla (esimerkiksi meteorologia) on hyvin kehittyneitä matemaattisia malleja joilla voidaan kuvata ennustettavaa ilmiötä ja käytettävissä on suurtietokoneita mallintamaan tapahtumia. Voidaan todeta, että mikäli jollain menetelmällä päästään arvaamista parempaan lopputulokseen, esimerkiksi seuraavan päivän tuotteiden kysynnän ennustamisessa, saatetaan menetelmästä saada taloudellista hyötyä.

Tässä työssä tutkitaan aikasarjoja ja niiden mallintamista. Erityisesti keskitytään aikasarjan mallintamiseen neuroverkolla. Keinoäly, eli AI (artificial intelligence) tekniikoihin luettavien neuroverkkojen käyttö on hitaasti lisääntymässä myös Suomessa, niitä ovat käyttäneet menestyksellisesti muutamat suuret teollisuusyritykset. Neuroverkot luetaan AI tekniikoihin sen vuoksi, että ne täyttävät kolme keinoälyn määritelmän kriteeriä (Haykin 1999: 34):

1. Kykyä tallentaa tietämystä
2. Kykyä soveltaa tietämystä ongelman ratkaisuun
3. Kykyä muuttaa tietämystä uuden kokemuksen perusteella

Tässä työssä käytetään MLP-neuroverkkoa, joka on yksi yleisimmin käytetyistä neuroverkoista. Tässä työssä myös sovelletaan kaikkia kolmea yllä mainittua keinoälyn määritelmän kriteeriä, tallennetaan aikasarjan malli opettamalla se neuroverkolle,

ennustetaan mallilla aikasarjan tulevia arvoja ja opetetaan neuroverkolle jatkuvasti aikasarjan mallia, ennustuksen edetessä.

Aikasarjat koostuvat tietyn muuttujan perättäisistä arvoista ajan kuluessa. Aikasarjat voivat olla jatkuva-aikaisia tai diskreettiaikaisia. Työssä rajoitutaan diskreettiaikaisiin aikasarjoihin, jolloin aikaa kuvaava indeksi t voidaan kuvata kokonaislukuna, joskin työssä kuvataan myös jatkuva-aikaisen aikasarjan muuntaminen diskreettiaikaiseksi, näytteistämällä. Lisäksi selvitetään aikasarjojen mallinnuksen perusteita ja perinteisiä metodeja. Työssä pohditaan myös sitä, milloin aikasarjasta on ylipäättään saatavissa ennusteita ja milloin on kysymys täysin satunnaisesta prosessista, joka sisältää vain valkoista kohinaa (white-noise) (Pynnönen 1999: 35). Mallinnettavana aikasarjana käytetään erään leipomoalan tuotteen kysyntää, vuoden 2007 eri viikonpäivinä.

Kyseessä on tuore päivittäistavaratuote, joten voidaan olettaa hyvistä ennusteista saatavan taloudellista hyötyä ainakin seuraavilla tavoilla. Tuotetun määrän ollessa riittävä, saa asiakas haluamansa tuotteet oikeana päivänä. Tämä on tärkeää erityisesti tuoreiden elintarvikkeiden kohdalla, sillä asiakasyrityksen saamatta jäänyt tuote on aina menetettyä myyntiä. Tuotteen loppukäyttäjä eli kuluttaja on jo tuossa vaiheessa päättänyt käyttää jotain korvaavaa tuotetta, joten tilannetta ei voida jälkitoimituksilla korjata. Lisäksi toimitusvarmuus vaikuttaa myös asiakasyrityksen luotettavuuskuvaan toimittajasta ja todennäköisesti se vaikuttaa myös ostojen määrään toimittajalta.

Toisaalta, jos tuotetaan liikaa, niin kaatopaikalle "kärätään" sekä tuotteeseen sitoutunutta työtä, että raaka-aineita. Tämä on ekologisesti kyseenalaista ja kun kysymyksessä on elintarvike, on se myös eettisesti kyseenalaista. Tosin puhtaasti taloudelliselta kannalta katsottuna on järkevämpää tuottaa hieman kysyntää enemmän, kuin liian vähän. Mikäli muutama tuote jää myymättä, on niihin sitoutunut pääoma suhteellisen pieni verrattuna saamatta jääneeseen katteeseen. Lisäksi on huomioitava toimitusvarmuuden heikkeneminen, jos muutama tuote jää toimittamatta.

1.1. Työn tavoite ja raja

Työn tarkoituksena on perehtyä aikasarjan mallintamiseen ja neuroverkkoihin, tuotteen kysyntää kuvaavan aikasarjan mallintamisen avulla. Työssä mallinnetaan erään päivittäistavaratuotteen kysyntää vuodelta 2007. Neuroverkon opetukseen käytetään 1.3.2004 – 31.8.2005 välisenä aikana kerättyä dataa. Saadulla mallilla ennustetaan kyseisen tuotteen kysyntää, syyskuussa 2007.

Tutkimusmenetelmänä tässä työssä käytetään teoreettis-empiiristä lähestymistapaa. Aikasarjoja koskevassa osuudessa käsitellään aikasarjan muodostavaa prosessia ja tutkitaan erilaisia malleja, joilla aikasarjoja voidaan kuvata. Neuroverkkoja käsittelevässä osassa käydään läpi neuroverkkojen perusteita ja paneudutaan yksityiskohtaisesti MLP-neuroverkkoon ja sen opettamiseen. MLP-neuroverkko on valittu siksi että sen käyttö on melko yleistä ja se soveltuu hyvin juuri aikasarjojen mallintamiseen. Lopuksi konstruoidaan MLP-neuroverkko, MS Excel ohjelmistoon VBA- koodilla.

Empiirisessä osiossa käytetään case-tutkimusmetodia. Tutkitaan millainen MLP-neuroverkko mallintaa annetun aikasarjan riittävän hyvin. Löydetyllä MLP-neuroverkolla tehdään lopuksi yhden askeleen pituisia ennusteita, yhden kuukauden jokaiselle myyntipäivälle.

Tutkimustulos koskettaa tätä tiettyä, tutkimuksen kohteena olevaa aikasarjaa, eikä ole siten suoraan yleistettävissä koskemaan muita aikasarjoja. Tulos antaa kuitenkin viitteitä siitä, miten hyvin MLP- neuroverkolla voidaan vastaavan kaltaisia aikasarjoja mallintaa.

Lopuksi analysoidaan ennustetulos, laskemalla erilaisia ennusteen luotettavuutta kuvaavia tunnuslukuja, kuten keskiarvoinen absoluuttinen poikkeama MAD (*mean absolute deviation*) ja keskiarvoinen neliövirhe MSE (*mean squared error*). Ennusteen tarkkuus on sitä parempi mitä pienemmät arvot, samalla aikavälillä, MAD ja MSE saavat. Lisäksi lasketaan keskiarvoinen absoluuttinen prosenttinvirhe MAPE (*mean*

absolute percentage error), joka kertoo sen kuinka suuria ennustevirheet ovat suhteessa todellisiin arvoihin.

Tutkimuskysymykset:

1. Millainen MLP- neuroverkko soveltuu aikasarjojen mallintamiseen?
2. Kuinka tarkkoja ovat yhden askeleen ennusteet?

2. TAUSTATEORIAM

2.1 Aikasarjat

Aikasarja kuvaa jonkin saman numeerisen suureen kehitystä ajassa t . Aikasarja muodostetaan mittaamalla halutusta prosessista suureen x arvoa ajan hetkillä $\{..., t_0, t+1, t+2, t+3, \dots\}$. Aikasarjan muodostava kaava on tällöin

$$x_{t_0}, x_{t_0+t}, x_{t_0+2t}, x_{t_0+3t}, \dots \quad (1)$$

missä x = suureen arvo

t = näytteiden välinen ajanjakso

Aikasarjoja voidaan muodostaa mistä tahansa sellaisesta prosessista, joka tuottaa mitattavia numeerisia arvoja, kuten esim. asiakasmäärät, tuotteen kysyntä, lämpötila, pörssikurssit jne. Mikäli aikasarjan yksittäinen arvo riippuu jossain määrin tai täysin sarjan edellisistä arvoista, niin silloin aikasarja sisältää informaatiota itsestään. Informaation avulla aikasarjaa analysoimalla voidaan tuottaa sellainen aikasarjan malli jolla voidaan ennustaa aikasarjan tulevia arvoja.

Aikasarjat voidaankin niiden ennustettavuuden perusteella jaotella stokastisiin ja deterministisiin aikasarjoihin. Aikasarja on *deterministinen* silloin kun ennustettava aikasarjan näyte määräytyy täysin edellisten arvojen perusteella. Tällöin aikasarjan arvo kullakin hetkellä voidaan ratkaista funktiosta joka kuvaa aikasarjan sen edellisistä arvoista. Aina ei ole selkeästi nähtävissä se, että aikasarja on deterministinen, tällaisia sarjoja voivat olla tietyt kaoottiset sarjat. Tutkittavat aikasarjat eivät kuitenkaan useinkaan ole deterministisiä. Aikasarjan syntyyn vaikuttavaan prosessiin liittyy deterministisen osan lisäksi jokin tuntematon *stokastinen* muuttuja jota ei pystytä ennustamaan. Aikasarjan koostuessa pelkästään *stokastisista* muuttujista sen ennustaminen on mahdotonta ja aikasarjaa kutsutaankin silloin valkoiseksi kohinaksi (*white-noise*). (Pynnönen 1999: 35.) Tällaista valkoista kohinaa on esimerkiksi analogisen TV:n "lumisade", tällöin ei myöskään ole löydettävissä autokorrelaatiota perättäisten suureiden suhteessa toisiinsa. Autokorrelaatio kertoo onko aikasarjan

näytteiden välistä korrelaatiota k viiveellä. Autokorrelaatiot r , eri viiveillä k , voidaan laskea seuraavalla kaavalla.

$$r_k = \frac{\sum_{t=1}^{N-k} (x_t - \bar{x})(x_{t+k} - \bar{x})}{\sum_{t=1}^N (x_t - \bar{x})^2}, \quad (2)$$

missä

r_k = autokorrelaatio viiveellä k ,

x_t = aikasarjan näytteen arvo ajan hetkellä t ,

\bar{x} = aikasarjan näytteiden keskiarvo (kaava 3),

N = aikasarjan näytteiden määrä.

Aikasarjat jaotellaan lisäksi niiden aikasekvenssin perusteella jatkuva-aikaisiin ja diskreettiaikaisiin aikasarjoihin. Aikasarja on *jatkuva-aikainen*, kun mitattavaa suuretta voidaan havainnoida jatkuvasti ajan suhteen. Tällöin suureesta saadaan mitattua arvo minä ajanhetkenä tahansa, tästä esimerkkinä analoginen radiolähetys. Tällöin aikasarjan aikaindeksi t , on reaaliluku.

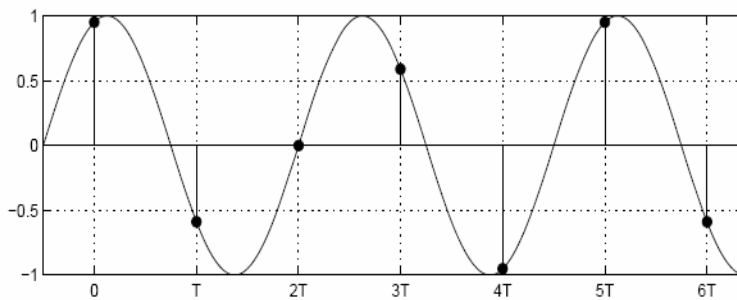
Diskreettiaikainen aikasarja sisältää mitatut suureen arvot, mitattuna aina ennalta määriteltynä, tiettyinä ajan hetkinä. Mittausten aikaväli on tasapituinen koska analyysiä on vaikea soveltaa satunnaisina hetkinä mitattuun dataan.

Ympäröivässä luonnossa esiintyvistä aikasarjoista valta osa on jatkuva-aikaisia, kuten valon määrä ja lämpötila eri vuorokauden aikoina. Aikasarjojen analysoinnissa tällaista jatkuva-aikaista aikasarjaa on kuitenkin hankala käsitellä tietokoneilla, siksi tutkittavat aikasarjat muunnetaan diskreettiaikaiseksi näytteistämällä tai summaamalla.

2.1.1 Näytteistys ja summaaminen

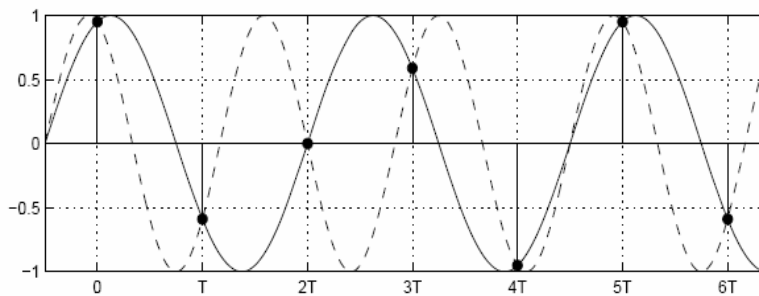
Näytteistys tarkoittaa sitä, että esimerkiksi A/D-muunnin (*analog/digital*) muuntaa jatkuva-aikaisen analogisen signaalin diskreettiaikaiseksi ja digitaalseksi. Näytteenotto tapahtuu aina jotain tiettyä taajuutta käyttäen, mitä suurempi taajuus sitä tarkempi malli alkuperäisestä signaalista saadaan. (Huttunen 2005: 1).

Näytteenottoteoreema (*sampling theorem*) kertoo mikä on riittävä näytteenottotaajuus signaalille. Näytteitä otetaan signaalista ajanhetkillä $0, T, 2T, 3T, 4T, \dots$ ja signaalin noina ajanhetkinä saamat arvot tallennetaan. Jatkuvasta signaalista käytetään merkintää $X_c(t)$ missä $t \in \mathbb{R}$, näytteistys tuloksena saadaan lukujono $X(n)$, jolle on voimassa ehto $X(n) = X_c(nT)$ missä $n = 0, 1, 2, 3, \dots$ ja T = kahden perättäisen näytteen aikaväli. (Huttunen 2005: 3).



Kuva 1. Näytteistys taajuudella T

Näytteenottotaajuus (*sampling frequency*) on vakion T käänteisluku $F_s = 1/T$. Mikäli näytteenottotaajuus on liian pieni, tapahtuu alinäytteistystä (*aliasing*), tällöin kahdella eri (kuvassa 2) signaalilla voi olla samat näytearvot. (Huttunen 2005: 4).



Kuva 2. Näytteistystaajuus on liian pieni.

"Jatkuva-aikainen signaali voidaan muodostaa uudelleen näytearvoistaan, jos näytteenottotaajuus f_s on vähintään kaksi kertaa niin suuri kuin signaalin sisältämä suurin taajuuskomponentti" (Huttunen 2005: 4).

Summaaminen tarkoittaa sitä, että esimerkiksi summataan tutkittavan nimikkeen kysyntä, päivä-, viikko-, kuukausi- tai vuositasolle, sen sijaan että käytettäisiin yksittäisiä asiakastilauksia.

2.1.2 Aikasarjojen ominaisuuksia

Aikasarjojen mallinnuksessa pyritään löytämään malli, joka kuvaa aikasarjan muodostavan prosessin siinä määrin, että voidaan mallin avulla joko ennustaa tulevia arvoja tai ymmärretään miten sarja muodostuu. Mallia luotaessa pyritään aikasarjasta laskemaan ja erottelemaan tiettyjä arvoja, joiden on havaittu vaikuttavan aikasarjan muodostumisprosessiin. Aikasarjojen ominaisuuksia ovat:

Stationäärisyys

Stationäärisyys tarkoittaa sitä, että aikasarjan tilastolliset ominaisuudet kuten:

Keskiarvo \bar{x} ,

$$\bar{x} = \frac{\sum_{i=1}^n x_i}{n}, \quad (3)$$

missä x_i = näytteen arvo indeksissä i ja

n = näytteiden määrä.

Keskihajonta S ,

$$S = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1}}. \quad (4)$$

Varianssi S^2

$$S^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1} \quad , \quad (5)$$

ja autokorrelaatio (kaava 2) ovat muuttumattomia tai muuttuvat vain vähän, ajan suhteen.

Trendi

Trendi on aikasarjan muodostavassa prosessissa ominaisuus, joka vaikuttaa pitkällä aikavälillä. Esimerkiksi tehtaan tuotantomäärä on pitkällä aikavälillä tarkasteltuna nouseva, jolloin trendi on siis nouseva. Lyhyissä aikasarjoissa on vaikeaa hahmottaa trendiä. Trendi voidaan joskus esittää suunnilleen suorana viivana, mutta sen muoto voi olla myös eksponentiaalinen tai vaikka S-käyrän muotoinen. Trendi voi johtua vaikkapa väestönkasvusta, muutoksissa elinolosuhteista, teknologisesta muutoksesta tai tuottavuuden lisääntymisestä. Aikasarjasta täytyy poistaa kausi- ja suhdannevaihtelu jotta trendi saadaan esiin.

Jotta aikasarjaa pystyttäisiin lineaarisella mallilla ennustamaan, on usein välttämätöntä poistaa trendi aikasarjasta. Trendin poistaminen aikasarjasta tapahtuu siten, että differoidaan, eli korvataan jokainen aikasarjan näyte x_t arvolla $x_t - x_{t-1}$, eli lasketaan erotus.

Suhdannevaihtelu

Suhdannevaihtelu voi olla vaikkapa muutaman vuoden mittaista, esimerkiksi yleisen taloudellisen tilanteen vaikutusta. Yleensä suhdannevaihtelu on hidasta ja vaikuttaa aikasarjaan lähes trendin tavoin ja näin ollen niitä voikin olla vaikea erottaa toisistaan.

Kausivaihtelu

Kausivaihtelu on samankaltaista kuin suhdannevaihtelu, mutta sen sykli on nopeampaa. Kausivaihteluita voi esiintyä esimerkiksi neljännesvuosittain, kuukausittain, viikoittain tai jopa päivittäin. Kausivaihtelut esiintyvät toistuvina jaksoina. Kausivaihteluiden syitä voivat olla mm. lämpötila, vuodenaika ja lomakaudet. Esimerkiksi vuodenajat vaikuttavat merkittävästi mm. kesä- tai talviurheiluvälineiden myyntiin, aiheuttaen kausivaihtelua.

Kausivaihtelu pyritään tilastoissa useimmiten poistamaan, vertaamalla mittaustulosta vuoden takaiseen samaan kyseiseen ajankohtaan, siis esimerkiksi vertaamalla tämän vuoden heinäkuun työttömyyslukua, heinäkuun työttömyyslukuun viime vuonna.

Jaksollisuus

Jaksollisuus esiintyy aikasarjassa säännöllisinä muutoksina tiettyinä ajanjaksoina. Jaksollisuudelle ei aina löydetä selkeää syytä, mutta esimerkiksi elintarvikemyymälän myynti eri vuorokaudenaikana ja viikonpäivänä muodostavat omat jaksonsa aikasarjassa. Jaksollisuus voidaan poistaa jakamalla aikasarjan arvo kunkin jakson keskiarvolla tai mikäli jakson pituus on tiedossa, voidaan laskea jakson pituuden päässä toisistaan olevien arvojen erotus, jokaiselle aikasarjan näytteelle $x_t - x_{t-l}$, missä l on jakson pituus.

Satunnaisvaihtelu

Satunnaisvaihtelu on se aikasarjan tuottaman prosessin osa, joka on nimensä mukaisesti satunnaista, eli se osaltaan aiheuttaa ennustemallin virheet. Aikasarjoja analysoitaessa pyritään edellä mainituille tekijöille löytämään mahdollisimman hyvin niitä kuvaavat mallit. Satunnaisvaihtelusta ei yleensä saada mitään irti, mutta se täytyy pystyä erottamaan kausivaihteluista, suhdannevaihteluista ja trendistä.

2.1.3 Ennustaminen ja aikahorisontit

Aikasarjan ennustamista varten luodaan sitä kuvaava matemaattinen malli. Tässä on otettava huomioon se, millaisella aikahorisontilla ennusteita on tarkoitus tehdä. Lähes poikkeuksetta on niin, että lyhyenajan ennusteet ovat tarkempia kuin pitkänajan ennusteet. Taloudellista toimintaa suunnitellaan moniin eri tarkoituksiin, siksi tarvitaan erilaisia ennusteita eri suunnitteluhorisonteille. Ennusteen aikahorisontin ja siihen perustuvan päätöksen tulisi aina kohdata. Mikäli siis ennuste tehdään seuraavalle päivälle, ei sen perusteella pidä tehdä päätöksiä seuraavalle viikolle. (Buffa & Sarin 1987: 54.)

Ennusteiden eri aikahorisontit riippuvat ennustettavasta muuttujasta ja siitä tarkoituksesta mihin ennuste tehdään. Ennustettavan aikasarjan aikafrekvenssi voi olla esimerkiksi muutamista millisekunneista useisiin vuosiin. Aikahorisontin pituus on näin ollen suhteellinen käsite. (Granger 1989: 6-7.)

2.1.4 Aikasarjojen mallinnusmenetelmiä

Aikasarjan mallintamiseen on useita eri menetelmiä kuten (Buffa 1987: 55):

- naiivit mallit
- klassinen aikasarja-analyysi
- liukuvan keskiarvon mallit
- autoregressiiviset mallit
- eksponentiaalisen tasoituksen mallit
- Box-Jenkins malli.

Seuraavaksi käydään näistä lävitse liukuvan keskiarvon malli, MA (*moving average*) ja klassinen aikasarja-analyysi, jotta saadaan hieman käsitystä siitä miten aikasarjoja perinteisesti on mallinnettu. Tässä työssä käytettyyn neuroverkkomalliin perehdytään kappaleessa 2.2. Neuroverkot.

Liukuvan keskiarvon mallit MA (moving average)

Liukuvan keskiarvon mallia voidaan käyttää silloin kun datassa ei ole nähtävissä trendiä tai kausivaihteluita. Mikäli datassa on trendi, niin MA malli laahaa trendin perässä. Liukuvan keskiarvon menetelmässä lasketaan ennuste aiempien toteutuneiden havaintojen keskiarvona. Liukuvan keskiarvon jaksojen määrä määrittää sen, kuinka monen aiemman havainnon keskiarvo lasketaan. Jos $k:n$ arvo on suuri, myös vanhemmat havainnot vaikuttavat ennusteeseen ja taas pienillä $k:n$ arvoilla vain tuoreimmat havainnot otetaan huomioon. MA mallin matemaattinen esitys on seuraava.

$$y = \frac{\sum_{i=1}^n x_{t-i}}{n}, \quad (6)$$

missä,
 y = tulevan jakson ennuste,
 n = keskiarvon laskennassa käytettävien jaksojen määrä ja
 x_{t-i} = aikasarjan aikaisemmat arvot ajanhetkillä $t-i$.

Liukuva keskiarvo voidaan laskea myös painotettuna, painottaen tällöin tuoreita havaintoja vanhoja havaintoja enemmän. Painotetun MA mallin matemaattinen esitys on seuraava.

$$y = \sum_{i=1}^n w_i x_i, \quad (7)$$

missä,
 y = tulevan jakson ennuste,
 n = keskiarvon laskennassa käytettävien jaksojen määrä,
 x_{t-i} = aikasarjan aikaisemmat arvot ajanhetkillä $t-i$,
 painokertoimet $w_1, w_2, \dots, w_n \geq 0$ ja $\sum w_i = 1$.

Klassinen aikasarja-analyysi

Klassisessa aikasarja-analyysissä (*decomposition methods for time-series*) yritetään mallintaa aikasarjan muodostavat komponentit, nämä komponentit vaikuttavat

ennusteeseen. Aikasarja ei enää tällöin ole stationäärinen, vaan sen keskiarvo ja hajonta muuttuvat ajan kuluessa. Kullekin komponentille hahmotellaan oma matemaattinen malli ja lopuksi mallit yhdistetään yhdeksi malliksi. Luotua mallia käyttäen voidaan aikasarjan tulevia arvoja ennustaa.

Klassista aikasarja-analyysiä voidaan käyttää tulevien arvojen ennustamiseen eripituisilla aikahorisonteilla. Klassista aikasarja-analyysiä voidaan ennustamisen lisäksi käyttää myös välineenä aikasarjojen käyttäytymisen ymmärtämiseen. Klassisten aikasarja-analyysien ennusteet eivät komponenttien erottelun vaikeudesta johtuen aina toimi kovin hyvin käytännössä. Komponentteihin pilkkominen voi kuitenkin auttaa hahmottamaan prosessiin vaikuttavat mekanismit ja löytämään piilevän kasvun tai laskun. (Hanke et al. 2001, 144.)

Klassisen aikasarjan näytteen arvo x , ajan hetkellä t , ajatellaan koostuvan neljästä (aiemmin esitellyistä) eri komponentista. Aikasarja voidaan tällöin esittää sen komponenttien summana.

$$X_t = T_t + C_t + S_t + R_t, \quad (8)$$

missä

X_t = aikasarjan arvo jaksolla t ,

T_t = trendi (trend) jaksolla t ,

C_t = suhdannevaihtelu(cycle) jaksolla t ,

S_t = kausivaihtelu(season) jaksolla t ,

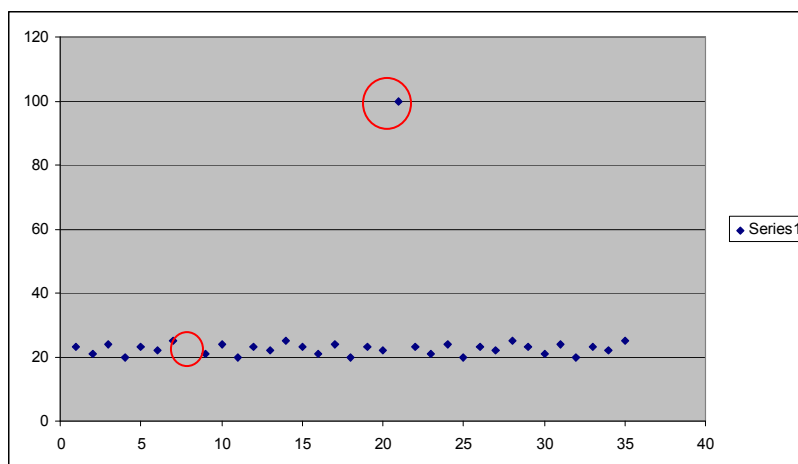
R_t = satunnaisvaihtelu (residual error) jaksolla t .

Edellä mainittujen menetelmien lisäksi on olemassa ns. *kausaalit menetelmät*. Nämä syy - ja seuraus menetelmät yrittävät mallintaa yhteyden eli syyn riippumattoman muuttujan ja/tai muuttujien suhteen. Esimerkkinä voitaisiin mainita mainoskampanja ja sen vaikutus myyntiin. Tällaisia muuttujia eivät klassiset aikasarja-analyysit ota huomioon. Kausaalisia malleja voidaan käyttää silloin kun on olemassa voimakkaita syy- ja seuraus suhteita, ja ne ovat tiedossa. Kausaalisia malleja ovat mm. regressiomallit, ekonometriset mallit ja neuroverkot. (Jain 2002, 19-20). Näistä neuroverkot tekevät poikkeuksen siinä mielessä, että niistä ei yksittäistä muuttujaa pystytä erottamaan. Tämä

johtuu siitä, ettei eri muuttujia voida erottaa, koska kaikki aikasarjaan tallentuva "tieto" on tallennettuna särmien painokertoimiin niin sanottuihin vapaisiin parametreihin.

2.1.5 Aikasarjadatan esikäsittely

Poikkeava näyte on aikasarjassa esiintyvä, ympäristöstään voimakkaasti poikkeava arvo (*outliner*). Se saattaa olla todellinen mittausarvo, joka syntyy poikkeuksellisissa olosuhteissa, tai mittaushäiriöstä johtuva virheellinen arvo. Poikkeavat näytteet saattavat vaikuttaa oleellisesti mallinnustulokseen, mikäli niiden vaikutusta ei huomioida. Lisäksi datassa saattaa olla "aukkoja" eli arvo puuttuu joltain tietyltä jaksolta, jostakin syystä. Yleensä aukot täytetään käyttämällä joko keskiarvoja tai interpoloimalla vastaava arvo. Kuvassa 3 on esimerkki tilanteesta, jossa on yksi muista täysin poikkeava arvo ja yksi puuttuva arvo.



Kuva 3. Esimerkki outlierista.

2.2 Neuroverkot

Tässä luvussa tutustutaan neuroverkkojen teoriaan ja perusteisiin. Neuroverkkojen yhteydessä puhutaan usein neurolaskennasta, joka on itse asiassa sama asia. Neuroverkkojen innoituksen lähteenä on ollut ihmisen aivojen toiminta ja niiden ”matkiminen”. Neuroverkko koostuu yleensä muutamasta neuronista, aina tuhansiin

neuroneihin. Neuroverkkoja toteutetaan sekä ohjelmallisesti, että erityisinä neurovirtapiireinä.

Neuroverkon arkkitehtuuriin vaikuttaa ratkaistavan ongelman luonne ja se mitä opetusalgoritmia käytetään. Neuroverkot voidaan jakaa kolmeen eri ryhmään, niiden arkkitehtuurin perusteella.

1. Yksikerros tai monikerros eteenpäin syöttö Perceptron -verkko (*Single/Multi -Layer Feedforward Networks*) on verkko jossa on yksi tai useampi taso, tällainen verkko tallentaa kuvauksen sivujen painokertoimiin, syötteestä vasteeseen. Perceptron verkko opetetaan valvotulla oppimisella.
2. Itse organisoituva kartta, SOM (*self organisation map*). Kohosen kehittämä SOM luo topologisen kuvauksen tutkittavasta datasta. SOM perustuu ohjaamattomaan oppimiseen.
3. Takaisin kytketty verkko (*Recurrent Networks*). Tunnetuin takaisin kytketty verkko lienee Hopfieldin-verkko, jota on käytetty mm. assosiatiivisena muistina ja optimointitehtävissä.

Neuroverkot soveltuvat erilaisiin tehtäviin niiden arkkitehtuurin mukaan. Neuroverkkojen soveltuvuus eri ongelmien ratkaisemiseen Tekesin raportin mukaan esitetään taulukossa 1. (Koikkalainen 1994.)

Taulukko 1. Neuroverkkojen soveltuvuus eri tehtäviin.

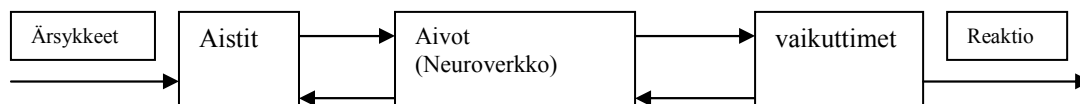
Neurolaskennan menetelmien soveltuvuus eräisiin tilasto- ja taloustehtäviin				
Neuroverkko- menetelmä	Tilastollinen mallinnus	Aikasarjat ja ennusteet	Tilastotiedon profilointi	Optimointi, esim. logistinen
Ohjattu oppiminen (MLP, Backprop)	●	●	⊗	⊗
Itseorganisaatio (SOM, ART...)	●	⊗	●	⊗
Tilansiirtoverkot (Hopfield,Bolzmnn)	⊗	⊗	—	●
● = hyvä soveltuvuus; ⊗ = soveltuu eräissä tilanteissa; — = ei sovellu				

Neuroverkoilla on mm. seuraavia hyödyllisiä ominaisuuksia (Haykin 1999: 2-6):

1. Epälineaarisuus. Mallinnettu neuroni voi olla joko lineaarinen tai epälineaarinen. Epälineaarisuus on tärkeä ominaisuus esimerkiksi puheen käsittelyssä.
2. Syöte-vaste kuvaus. Neuroverkkoa opetetaan ohjatusti, satunnaisilla näytteillä kohde datasta (painokertoimia muuttamalla), kunnes syöteen ja vasteen välinen ero on haluttu. Opetuksen jälkeen annettu samankaltainen syöte antaa saman vasteen eli neuroverkko kykenee yleistämään oppimansa ja muodostaa syöte-vaste kuvauksen verkkoon. Tämä on luokittelussa käyttökelpoinen ominaisuus.
3. Mukautuminen. Neuroverkoissa on sisäänrakennettu kyky säätää painokertoimia ympäristön mukaan. Tämä mahdollistaa sen, että johonkin tilanteeseen opetettu neuroverkko voidaan helposti opettaa uuteen muuttuneeseen tilanteeseen. Tämä ominaisuus tekee neuroverkoista käyttökelpoisen työkalun hahmontunnistukseen, signaalinkäsittelyyn ja erilaisiin säätötehtäviin.
4. Neuroverkot ovat virheensietokykyisiä. Kuten esikuvansa aivot, nekin sietävät datassa kohinaa.
5. Neurobiologinen analogia. Neuroverkkojen malleilla on pyritty jäljittelemään luonnollista toimintaa.

2.2.1 Neuroverkkojen esikuva

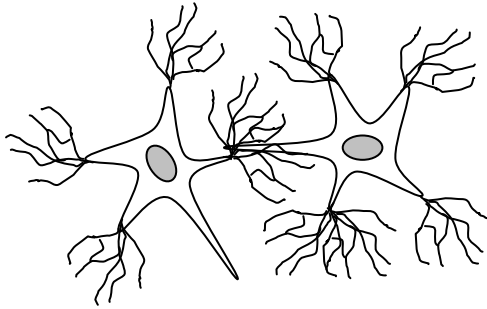
Ihmisen hermojärjestelmä voidaan kuvata kolmen tason järjestelmänä. Aivot toimivat järjestelmän keskuksena, ottaen jatkuvasti vastaan informaatiota aistinelimiltä ja tehden päätöksiä siitä miten informaatioon reagoidaan (Haykin 1999: 6).



Kuva 4. Ihmisen hermojärjestelmä.

Aivot ovat neuroverkkojen esikuva. Ihmisen aivot käsittelevät tietoa eri tavalla ja tehokkaammin kuin tietokone. Aivot oppivat kokemuksesta ja tallentavat tietoa

neuronien välisiin yhteyksiin eli synapseihin. Tätä oppimista pyritään neuroverkoilla mallintamaan. Aivot muuttavat ja vahvistavat yhteyksiä "tulittamalla" (*fires*) aksonien välityksellä. Yhteydet voivat olla joko sähköisiä tai kemiallisia impulsseja. (Haykin 1999: 1-2).



Kuva 5. Aivosolut ja niistä lähtevät aksonit.

"Aivot ottavat vastaan, varastoivat, käsittelevät ja tuottavat informaatiota – ja tietävät tekevänsä niin. Ihmisaivoissa tähän työhön on valjastettu noin 100 miljardia hermosolua, jotka muodostavat tiettävästi yhtenäisen verkon. Hermosoluilla eli neuroneilla on seuraavia tehtäviä:

- 1) ottaa vastaan muista soluista tulevia impulsseja ja "laskea" oma signaali eli muihin neuroneihin lähetettävä viesti sisään tulevan informaation funktiona,*
- 2) kytkentöjen voimakkuuksia ja solun sisäistä tilaa vastaanotettujen muita apurakenteita kuten kalvostoja, nestetiloja ja verisuonia"*

<http://www.biomag.hus.fi/braincourse/luentomoniste2001.html>

Ihmisaivojen tehokkuutta kuvaa hyvin se, että niiden laskentakyky perustuu arviolta 10 biljoonan aivosolun ja niiden noin 60 triljoonan synapsin yhteyksiin. Energiaa aivot kuluttavat vain noin 10 -16 joulea / operaatio / sekunnissa. (Haykin 1999:6).

Erityisesti hahmontunnistuksessa aivot ovat lyömättömät. Esimerkiksi oheisessa kuvassa on riittävästi informaatiota ihmisaivoille. Kyseessä on maata haistelevan Dalmatiankoiran hahmo, kuvattuna vasemmalta sivulta. Kuvan 6 sisältämän kompleksisen informaation kuvaamiseen algoritmiksi tietokoneelle ei toistaiseksi ole

ratkaisua. Miten kuvata mitkä mustat läikät kuuluvat koiraan, mitkä muuhun ympäristöön? (Freeman & Skapura 1992: 2).



Kuva 6. Dalmatiankoiran hahmo (Freeman 1992: 3).

Selvää on, että nykyisillä tietokoneilla ei voida ihmisaivojen kaltaista massiivista rakennetta toteuttaa. Parhaimmillaankin, suurimpiin kuluvalle supertietokoneella, pystytään simuloimaan hiiren ajattelua n.10 sekuntia. Jotkut tekoälyn tutkijat ovat sitä mieltä, ettei koskaan tulevaisuudessakaan päästä lähellekään ihmisaivojen veroista konetta.

"Käytännön tekoälytutkimus ei enää haaveile rakentavansa ihmisenkaltaista älykästä robottia, vaan tavoitteet ovat vaatimattomammat. Tekoälyn voisi sanoa olevan tietorakenteiden ja algoritmien kehitystä tietämyksen esittämiseen, pätevän päättelyn suorittamiseen sekä älykkään järjestelmän tavoitteiden kannalta parhaiden toimintojen valitsemiseen."

(Kokkarinen & Ala-Mutka 2002: 305).

2.2.2 Neuroverkon rakenne

Edellä kuvattua informaation siirtoa ja yhteyksiä pyritään siis keinotekoisilla neuroverkoilla jäljittelemään. Verkko muodostuu keinotekoisista neuroneista ja niiden aktivaatiofunktioista sekä särmistä ja niiden painokertoimista. Laskenta tapahtuu siten, että syöte annetaan piiloneuroneille painotettuna summana, niiden ulostulo toimii seuraavan kerroksen syötteenä ja viimeiseltä kerrokselta saadaan koko verkon vaste.

Neuronin tarkoitus on mallintaa aivosolun toimintaa. Yksittäinen neuroni on ”tyhmä”, se osaa vain ottaa vastaan painotettuna syötteen ja antaa ulos siihen liitetyn aktivaatiofunktion arvon. Keinotekoisen neuronin (*perceptron*) malli (Haykin 1999: 11):

$$y_k = \varphi\left(\sum_{j=0}^m w_{kj} x_j\right), \quad (9)$$

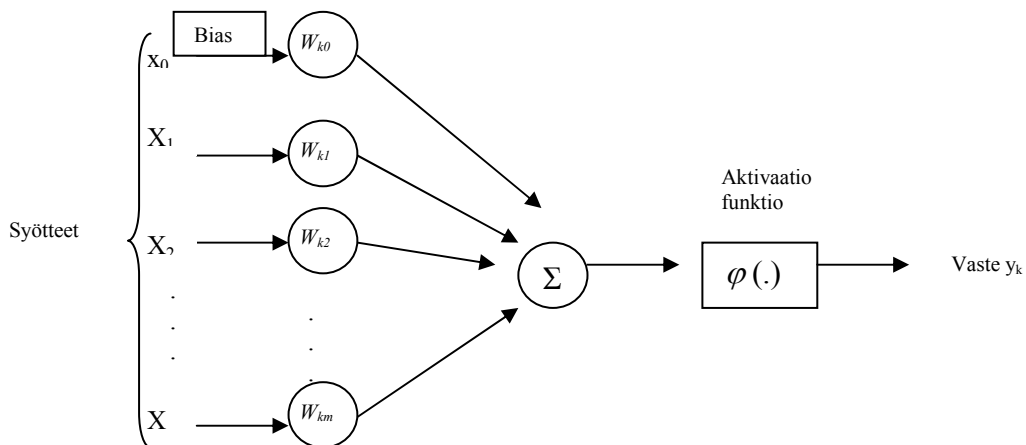
missä,

y_k = neuronin k vaste,

$\varphi(.)$ = neuronin aktivaatio funktio,

x_1, x_2, \dots, x_m = neuronin saamat syötteen ,

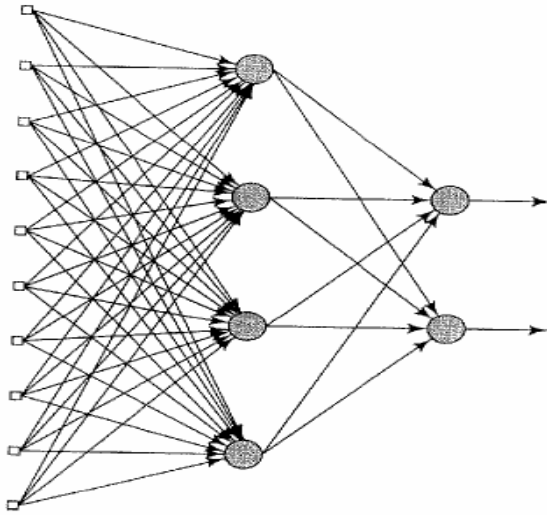
$w_{k1}, w_{k2}, \dots, w_{km}$ särmien m liitetty painokertoimet.



Kuva 7. Keinotekoisen neuronin malli (syötteen, neuroni ja vaste).

Järjestettäessä näitä neuroneita rinnakkain ja peräkkäin, alkaa niiden yhteistuloksesta syntyä "järkeä". Yksikerroksisella verkolla voidaan mm. tehdä lineaarista luokittelua, mutta se ei pysty mallintamaan epälineaarisia tapahtumia. Kun verkkoon lisätään vähintäänkin yksi lisäkerros ja aktivaatiofunktio on epälineaarinen, niin verkko kykenee oppimaan epälineaarisia malleja.

Kun kuvan 8 mukaisia neuroneita asetetaan vähintään kaksi kerrosta ja rinnakkain, saadaan niin sanottu monikerrosneuroverkko MLP (*multi layer perceptron*).



Syötoneuronit Piiloneuronit Vastoneuronit

Kuva 8. Monikerrosneuroverkko.

MLP-verkko on kuvaus $\mathbf{y} = F(\mathbf{w}, \mathbf{x})$, joka muodostuu sisääntulon \mathbf{x} ja ulostulon \mathbf{y} välillä painokertoimien \mathbf{w} avulla. Tämä kuvaus on epälineaarinen funktio. MLP-verkko, jossa on yksi piilokerros, voidaan matemaattisesti esittää kaavalla:

$$y_k = f\left(w_{ok}^2 + \sum_{j=1}^q w_{jk}^2 f\left(w_{oj}^1 + \sum_{i=1}^p w_{ij}^1 x_i\right)\right), \quad (10)$$

missä

$y_k = k:n$ en ulostulo neuronin vaste,

w_{ok}^2 = ulostuloneuronien särmiin liitetyt painokertoimet,

w_{oj} = piiloneuronien särmiin liitetyt painokertoimet,
 X_1, X_2, \dots, X_m = verkon saamat syötteet .

Huom! Yläindeksi w_{ok}^2 ei ole potenssi vaan kertoo verkon neuronin tason.

Verkolle annetaan syöte suoraan syöteneuroneihin yksi kerrallaan. Syöteneuroneissa ei ole toiminnallisuutta, ne vain vastaanottavat tiedon. Tieto siirtyy painotettuna summana piiloneuroneille. Piiloneuroneiden reagointi vasteeseen riippuu niiden aktivaatiofunktioista. Viimeiseltä neuronin kerrokselta saatu vaste on verkon lopullinen vaste.

Opetusvaiheessa lasketaan verkon virhe. Virhe syötetään takaisin verkkoon päinvastaisessa järjestyksessä ja painokertoimia päivitetään lasketun virheen mukaan kunnes verkon virhe on joko minimoitunut tai haluttu virhetaso saavutettu. Tätä selvitetään tarkemmin luvussa 2.2.4 neuroverkon opettaminen.

Aktivaatiofunktio on neuronin "sisällä" ja vaikuttaa siihen miten saatuun syötteeseen reagoidaan. Syötteiden painotetun summan ja aktivaatiofunktion antama tulos toimii seuraavan kerroksen syötteenä ja viimeisellä vastetasolla niiden yhteensä tuottama tulos on siis verkon vaste.

Kolme eniten käytettyä aktivaatiofunktioita (Haykin 1999):

Threshold

$$\varphi(v) = \begin{cases} 1 & \text{jos } v \geq 0 \\ 0 & \text{jos } v < 0 \end{cases} \quad (11)$$

missä v = syötteiden painotettu summa.

Threshold funktiota käytetään yksikerrosverkoissa.

Binäärinen sigmoidi

$$\varphi(v) = \frac{1}{1 + e^{-av}}, \quad (12)$$

missä
 v = syötteiden painotettu summa ja
 a = muutosnopeusparametri.

Binäärinen sigmoidi antaa arvon välille 0 – 1, se soveltuu mm. sellaisiin verkkoihin, jossa mallinnettava data on epälineaarista, eikä saa negatiivisia arvoja (esim. tuotteen kysyntä eri viikonpäivinä).

Bipolaarinen sigmoidi

$$\varphi(v) = \frac{1 - e^{-av}}{1 + e^{-av}}, \quad (13)$$

missä
 v = syötteiden painotettu summa ja
 a = muutosnopeusparametri.

Bipolaarinen sigmoidi antaa arvon väliltä -1 – 1 ja soveltuu sellaisiin verkkoihin joissa mallinnettava data on epälineaarista ja voi saada myös negatiivisia arvoja.

2.2.4 Neuroverkon opettaminen

Neuroverkkoa voidaan opettaa sekä ohjatulla että ohjaamattomalla oppimisella. *Ohjaamattomaksi oppimiseksi* kutsutaan tilannetta, jossa verkolle ei "näytetä" oikeaa tulosta vaan sen on löydettävä itse tasapainotila. Tällaista ohjaamatonta oppimista käytetään esimerkiksi SOM verkoissa siten, että verkolle "näytetään" opeteltavasta datasta näytevektori. Sitä euklidisesti lähinnä olevan neuronin särmiä painotetaan lisää.

Tässä työssä käytetään *ohjattua oppimista*. Opettajalla on tässä tapauksessa tietoa aikasarjasta, jota neuroverkon tulee mallintaa. Verkolle opetetaan syöte-vaste-

esimerkkien avulla aikasarjan malli. Annettujen syöte-vaste-esimerkkien perusteella verkko muuttaa särmiin liittyviä painokertoimiaan opetusalgoritminsa mukaisesti, kunnes virhe on riittävän pieni tai opettaja määrää keskeyttämään. Tämän jälkeen verkon on suoriuduttava tulevista näytteistä ilman että niitä on aiemmin opeteltu. (Haykin 1999: 63-64 .)

Neuroverkon opettamisen vaiheet:

1. Asetetaan särmien painoille satunnaiset alkuarvot.
2. Syötetään verkolle opetusjoukosta alkioita.
3. Lasketaan verkon vaste annetulle syötteelle.
4. Lasketaan virhe, vaste – tavoite vaste.
5. Korjataan särmien painokertoimia virhettä vastaavasti.
6. Kun kohdat 1-2 on toistettu niin usein, ettei virheen havaita enää pienenevän, voidaan todeta, että verkko on opetettu riittävän hyvin.
7. Testataan neuroverkon suorituskkyä testijoukolla (eri pistevierasjoukko testiaineistolle). Mikäli ennustuskky on liian heikko, tarkastellaan piiloneuroneiden määrää (lisää vai vähemmän) ja aloitetaan vaiheesta 1.

Back-propagation opetus-algoritmi

Seuraavassa käydään MLP-verkoissa yleisesti käytetty back-propagation opetus-algoritmi tarkemmin läpi vaihe vaiheelta. (Zurada 1992: 188:190.). Yläindeksi kaavoissa tarkoittaa neuronin tasoa verkossa, 0 on vasteneuroni ja 1 on piiloneuroni.

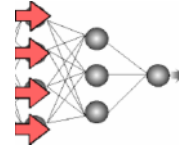
Askel 1 Alustetaan neuronien särmille satunnaiset pienet painokertoimet.

Esimerkki on otettu tässä työssä käytetystä sovelluksesta.

```
'alustetaan pienillä satunnaisluvuilla hide_out väliset painot
For i = 1 To hide_node + 1
For j = 1 To out_node
    Sheet5.Cells(i, j).Value = (1.5 * Rnd) - 0.5
Next j
Next i
```

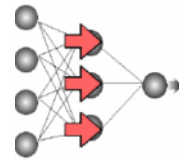
Askel 2 Syötetään syöteneuroneihin näyte vektorin \mathbf{Z} arvot. Lasketaan vasteet kultakin piiloneuronilta, syötetään ne vastaneuronille syötteenä ja lasketaan sen vaste. Yläindeksi kaavoissa tarkoittaa neuronin tasoa verkossa, 0 on vastaneuroni ja 1 on piiloneuroni.

Syötetään näytevektori \mathbf{Z} neuroverkolle



Lasketaan piiloneuroneiden vasteet

$$y_k^1 = \varphi\left(\sum_{j=0}^m w_j^1 x_j\right),$$



(14)

missä

y_k^1 = k:nen piiloneuronin vaste,

φ = neuronin aktivaatio funktio,

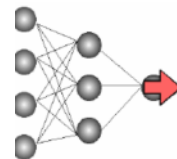
w_j^1 = piiloneuronin j:nenteen särmään liitetty paino,

x_j = syötevektorin \mathbf{Z} j:nes arvo,

m = syötevektorin \mathbf{Z} arvojen määrä (dimensio).

Lasketaan vastaneuroneiden vasteet

$$y_k^0 = \varphi\left(\sum_{j=0}^m w_j^0 y_j^1\right),$$



(15)

missä

y_k^0 = k:nen vastaneuronin vaste,

φ = neuronin aktivaatio funktio,

w_j^0 = vastaneuronin j:nenteen särmään liitetty painokerroin,

y_j^1 = j:nen piiloneuronin vaste,

m = piiloneuroneiden määrä.

Askel 3 Lasketaan verkon virhe E , joka on puoli kertaa neuroverkon vasteiden virheiden summa.

$$E \leftarrow E + \frac{1}{2} \|d - y^0\|^2, \quad (16)$$

missä

E = neuroverkon vasteiden virheiden neliöiden summa,

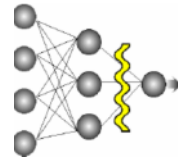
d = haluttu vaste,

y^0 = ulostuloneuronin laskettu vaste.

Askel 4 Lasketaan neuroneiden vasteesta ja halutusta vasteesta virhetermit kullekin neuroverkon tasolle.

Lasketaan virhetermit ulostuloneuroneille.

$$\delta_k^0 = (d_k - y_k^0)(1 - y_k^0)y_k^0, \quad (17)$$



missä

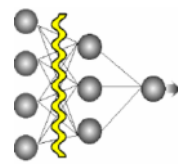
δ_k^0 = k:nen vastaneuronin virhetermi,

d_k = k:nen vastaneuronin haluttu vaste,

y_k^0 = k:nen vastaneuronin vaste.

Lasketaan virhetermit piiloneuroneille

$$\delta_k^1 = y_k^1 (1 - y_k^1) \sum_{k=1}^k \delta_k^0 w_k^0, \quad (18)$$



missä

δ_k^1 = k:nen piiloneuronin virhetermi,

y_k^1 = k:nen piiloneuronin vaste,

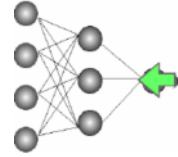
k = piiloneuronien määrä,

δ_k^0 = k:nen vastaneuronin virhetermi,

w_k^0 = k:nen vastaneuronin ja piiloneuronin välisen särmän painokerroin.

Askel 5 Päivitetään vastaneuronien särmien painokertoimia saadun virhetermin perusteella

$$W_k^0 \leftarrow W_k^0 + \eta \delta_k^0 y_k^0, \quad (19)$$



missä

W_k^0 = k:nen vastaneuronin ja piiloneuronin välisen särmän painokerroin,

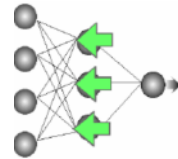
η = opetusnopeus parametri,

δ_k^0 = k:nen vastaneuronin virhetermi,

y_k^0 = k:nen vastaneuronin vaste.

Askel 6 Päivitetään piiloneuronien särmien painokertoimia saadun virhetermin perusteella

$$W_k^1 \leftarrow W_k^1 + \eta \delta_k^1 y_k^1, \quad (20)$$



missä

W_k^1 = k:nen piiloneuronin ja syöteneuronin välisen särmän painokerroin,

η = opetusnopeus parametri,

δ_k^1 = k:nen vastaneuronin virhetermi,

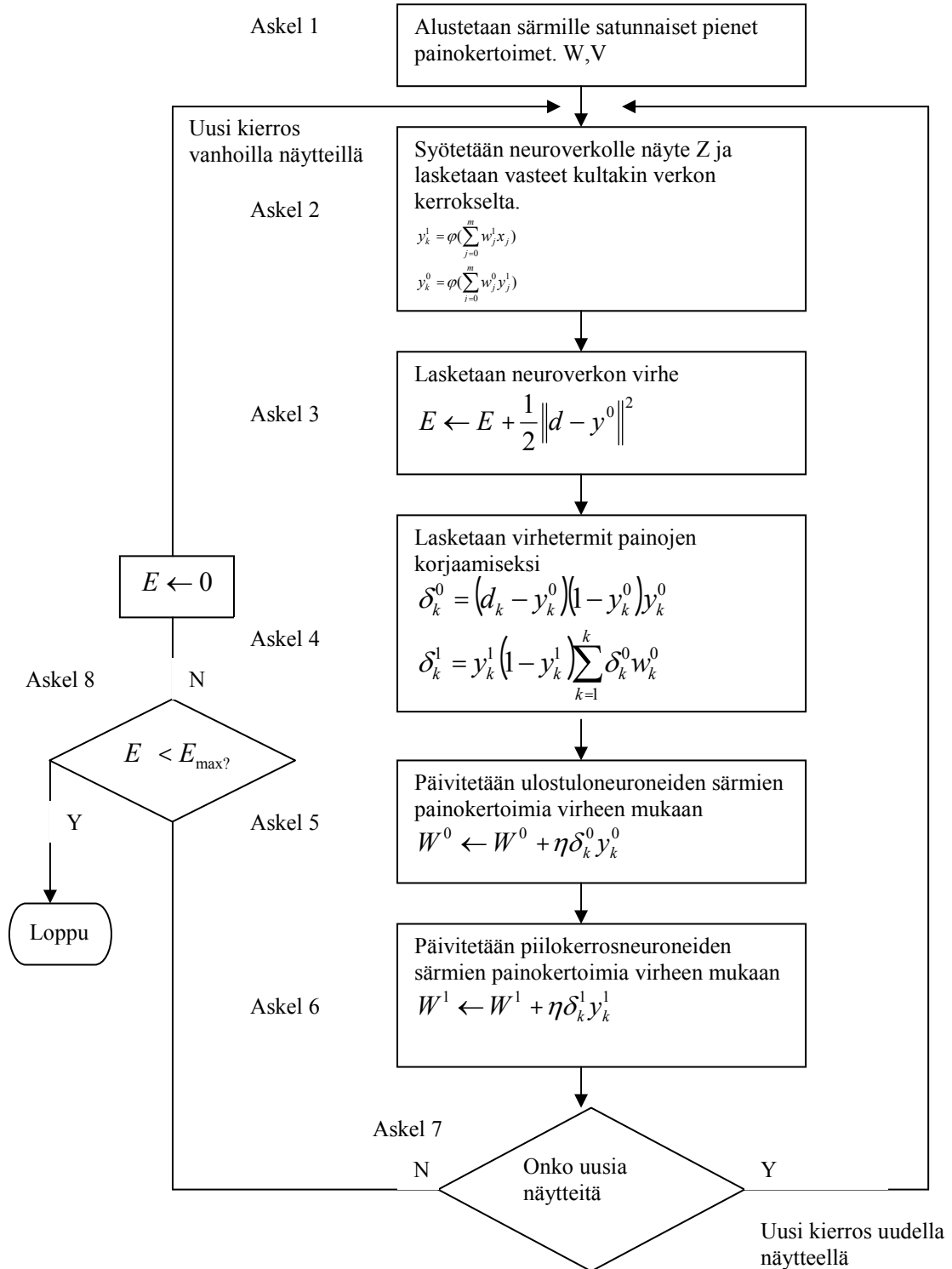
y_k^1 = k:nen vastaneuronin vaste.

Askel 7 Tarkistetaan onko näytevektoreita Z olemassa lisää, jos on, syötetään uusi näytevektori Z neuroverkolle, jos ei ole siirrytään seuraavaan vaiheeseen.

Askel 8 Tarkistetaan onko neuroverkon virhe pienentynyt, jos on, mutta se ei ole nolla jatketaan opetusta syöttämällä näytevektorit Z uudelleen neuroverkkoon.

Mikäli virhe ei enää pienene tai se on halutulla tasolla, päätetään algoritmi.

Back-propagation opetus-algoritmi kaaviona. (Zurada 1992: 187.)



2.2.5 Neuroverkon rakenteen suunnitteleminen

Monikerrosneuroverkko koostuu nimensä mukaisesti useammasta kuin yhdestä kerroksesta, yleensä ei kuitenkaan tarvita kahta kerrosta enempää tasoja. Aikasarjan kompleksisuuden lisääntyessä voidaan lisätä piilokerroksen neuroneiden määrää, jolloin lisätasoja ei tarvita (Masters 1993: 174).

Monissa luokittelutehtävissä riittää yksikin taso, mutta silloin lajitteluongelman täytyy olla lineaarisesti separoituva. Liian vähäinen neuronien määrä piilokerroksessa estää verkkoa oppimasta ongelmaa, toisaalta liiallinen neuroneiden määrä hidastaa oppimista ja voi aiheuttaa ylisovittamista. Ylisovittaminen tarkoittaa tilannetta, jossa opetusjoukon koko on niin pieni neuroneiden särmien painojen määrään nähden, että verkko oppii koko opetusjoukon yksityiskohtaisesti ja menettää tässä tapauksessa yleistämiskykynsä. (Koikkalainen 1994.)

Yleistämiskyky tarkoittaa sitä, että verkko pystyy soveltamaan oppimaansa mallia uuteen, ennennäkemättömään dataan ja antaa siitä huolimatta järkevän vasteen. Kun neuroverkko on ylisovitettu, tunnistaa se opetusjoukon alkiot erinomaisesti, mutta uusien alkioden tunnistaminen epäonnistuu. (Koikkalainen 1994.)

Karkeana sääntönä piiloneuroneiden määrälle voidaan pitää sitä, että painokertoimien määrä ei saisi ylittää syötoneuronien määrää kerrottuna aikasarjan näytteiden määrällä. Piilokerroksen neuronien määrän tulee olla aikasarjan näytteiden määrää pienempi. Mikäli tehtävä vaatii kaarevia päätöspintoja, on piilokerroksen neuronien määrän oltava noin kolme kertaa syötoneuronien määrä. (Koikkalainen 1994.)

Oikean piilokerroksen koon löytämiseksi täytyy siis kokeilla mikä on sopiva piiloneuroneiden määrä. Voidaan aloittaa pienellä piilokerroksella ja lisätä piilokerroksen neuroneiden määrää kunnes verkko on oppinut opetusdatan riittävän hyvin.

2.2.6 Datan esi- ja jälkikäsittely

Neuroverkko ei voi oppia dataa missä muodossa tahansa ja esitysmuoto vaikuttaa myös oppimismenoon (Haykin 1999: 181). Datan esi- ja jälkikäsittelyn toteuttamiseen ei ole mitään yleispätevää ohjetta, esitysmuoto täytyykin aina tapauskohtaisesti ratkaista ja tavoite on saada data muotoon, joka sopii parhaiten valitulle neuroverkkoarkkitehtuurille. (Koikkalainen 1994.)

Tässä työssä käytetään aktivaatiofunktiona binääristä sigmoidia. Binäärinen sigmoidi saa arvoja nolasta yhteen. Mikäli mitään käsittelyä ei datalle tehdä, verkon vasteelle ei saada mitään tulkintaan. Mitä esimerkiksi tarkoittaa jos neuroverkko ennustaa seuraavan päivän myynnin määräksi 0,128828? Jotta sigmoidin antama tulos olisi ymmärrettävässä muodossa, tulee aikasarjan kaikki arvot skaalata välille [0,1] seuraavalla kaavalla.

$$x' = \frac{x_i}{S}, \quad (21)$$

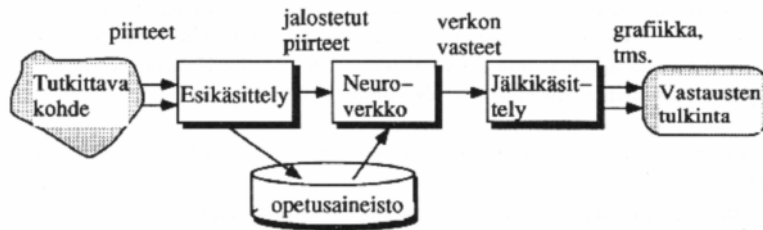
missä
 x' = skaalattu aikasarjan arvo,
 x_i = alkuperäinen aikasarjan arvo ja
 S = aikasarjan keskihajonta (kaava 4).

Jos aikasarjaan sopii paremmin bipolaarinen sigmoidi, skaalaukseen voidaan käyttää kaavaa,

$$x' = \frac{x_i - \bar{x}_i}{S}, \quad (22)$$

missä
 x' = skaalattu aikasarjan arvo,
 x_i = alkuperäinen aikasarjan arvo,
 \bar{x}_i = aikasarjan keskiarvo ja
 S = aikasarjan keskihajonta (kaava 4).

Vastaavalla tavalla neuroverkon vaste jälkikäsittellään edellä esitetyillä kaavoilla, käänteisesti.



Kuva 9. Datan esi- ja jälkikäsittely.

2.2.7 VBA- Excel neuroverkko sovellus

Neuroverkon sovellus on toteutettu MS Excel ohjelmaan. Esisijaisena tavoitteena on tutustustua tarkemmin neuroverkon toteutukseen käytännössä koodin kautta, siten että ymmärrys kokonaisuudesta muodostuisi mahdollisimman hyväksi. Sovelluksen opetus toteutettiin Zurada:n (1992) kirjassaan esittämä back-propagation-algoritmin mukaisesti.

Toteutus alustan valintaan vaikutti se, että MS Excel on lähes jokaisessa Windows pohjaisessa tietokoneessa ja sovellus voitiin toteuttaa nopeasti kyseiseen ympäristöön. Lisäksi käyttöliittymä ja datan visualisointi ominaisuudet olivat lähes valmiina ja aikasarjadatan tarkastelu ja särmien painokertoimien tarkastelu on helppoa valmiissa käyttöliittymässä.

Oppimistarkoituksessa on myös hyvä, että neuroverkon käyttämät särmienpainokertoimet ovat tarkasteltavissa ja tarvittaessa suoraan muokattavissa. Testaamisen näkökulmasta tarkasteltuna on myös tärkeää, että sovelluksen neuroverkon syöte-, piilo- ja vasteneuroneiden määrä on säädettävissä.

Neuroverkkosovelluksen vaatimukset:

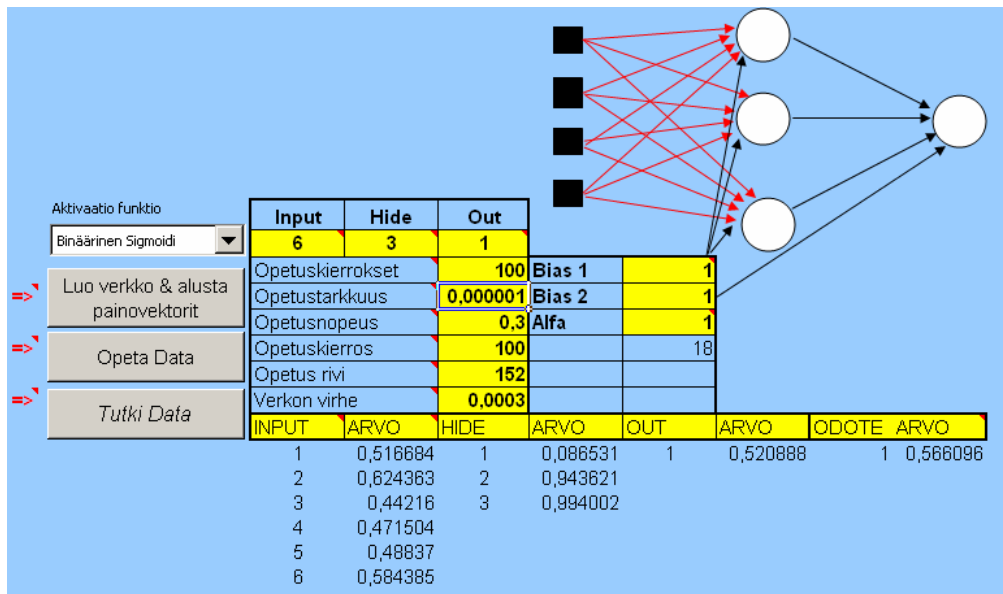
1. Syötevektorin avaruus voi olla 1, 2, 3, ... n. eli syöteuroneiden määrä pitää olla säädettävissä
2. Yksi piilokerros, jonka piiloneuroneiden määrä pitää olla säädettävissä

3. Yksi vastaneuroni. Luotaessa ennusteita aikasarjan perusteella yhdelle päivälle yksi askel kerrallaan, ei ole tarvetta useammalle vasteelle.
4. Käytetään aktivaatiofunktiona singmoidia, joka saa arvoja väliltä 0... 1, koska aikasarjadata ei saa negatiivisia arvoja.
5. Aktivaatiofunktion muutosnopeusparametri on säädettävissä.
6. Opetuksessa noudatetaan back-propagation algoritmia.
7. Opetuskierrosten määrää on voitava säätää.

2.2.8 VBA- Excel neuroverkko sovelluksen käyttöliittymä

Käytettäessä VBA-Excel neuroverkkosovellusta on ensin määriteltävä syöte-piilo- ja vastaneuronien määrä, sekä käytettävä aktivaatiofunktio. Sovelluksessa on käytettävissä sekä binäärinen että dipolaarinen sigmoidi funktio. *Luo verkko & alusta painovektorit-painike* luo sovelluksen tarvitsemat taulukkomuuttujat sekä valitsee pienet satunnaisluvut särmien painokertoimiksi (kuva 10). Särmien painokertoimet tallennetaan omille välilehdilleen.

Kun verkko on luotu, voidaan valita haluttu opetuskierrosten määrä, opetustarkkuus, opetusnopeus sekä alfa. *Alfa* vaikuttaa funktion herkkyyteen eli siihen miten jyrkkä sigmoidi- funktion kuvaajan käyrä on. *Opetusnopeus* kertoo kuinka voimakkaasti verkon virhettä korjataan muuttamalla särmien painokertoimia.



Kuva 10. VBA- Excel sovelluksen käyttöliittymä

Opeta Data- painiketta painettaessa sovellus käy läpi annettuja datanäytteitä, päivittäen jokaisen näytteen jälkeen särmien painokertoimia. *Opetusrivi* kertoo miltä riviltä, *opetusdata*-välilehdeltä (kuva 11) sillä hetkellä luetaan ja *opetuskierron*-kenttä kertoo, opetusdatakierroksen senhetkisen lukumäärän. Opetusdata näytteet tulee olla syötettynä (ja esikäsiteltyinä) *opetusdata*-välilehdelle.

B	C	D	E	F	G	H	I
INPUT1	INPUT2	INPUT3	INPUT4	INPUT5	INPUT6	ODOTE1	OUT1
0,45241	0,57617	0,47142	0,62011	0,42333	0,42052	0,41829	0,48363
0,576172	0,47142	0,62011	0,42333	0,42052	0,41829	0,52325	0,5365

Kuva 11. VBA- Excel sovelluksen opetusdata

Tutki Data- painike käynnistää opetuksen jälkeen sovelluksen lukemaan datanäytteitä *data*-välilehdeltä. Sovellus tuottaa jokaiselle datanäytteelle sitä vastaavan neuroverkon vasteen, joka tallennetaan näytettä vastaavalle riville sarakkeeseen OUTx (kuva 12). Datanäytteiden tulee olla neuroverkolle ennennäkemättömiä, jos käytetään samaa dataa kuin opetusdatassa niin vastekin on täysin sama.

A	B	C	D	E	F	G	H
RMI	INPUT1	INPUT2	INPUT3	INPUT4	INPUT5	INPUT6	OUT1
1	0,455474	0,599179	0,42343	0,414964	0,424001	0,566096	0,486216
2	0,599179	0,42343	0,414964	0,424001	0,566096	0,486216	0,621357

Kuva 12. VBA- Excel sovelluksen tulodata

OUTx sarakkeesta voidaan siis lukea kutakin näytedataa vastaava ennuste. Liitteessä 1 on sovelluksen VBA- koodi kokonaisuudessaan.

2.3 Aikasarjamallin ennustuskyvyn ja ennustevirheen mittaaminen

Ennustevirhe on toteutuneen arvon ja ennustetun arvon välinen erotus. Ennusteissa on aina virheitä, paitsi jos ennustettava aikasarja on deterministinen ja sitä kuvaava malli on muodostettu oikein. Ennustevirheet voivat olla datassa olevasta virheestä (näppäilyvirhe, viallinen anturi, tiedonsiirtovirhe jne.), ennustemallin muodostamisessa tapahtuneesta virheestä (systemaattista) ja aikasarjan satunnaisuudesta johtuvia virhettä. Satunnaisvaihteluista johtuva virhe on mukana kaikissa aikasarjoissa. Satunnaisvaihtelu on se osa aikasarjasta, jota ei saada mallinnettua. Ennustevirheen mittaaminen antaa tietoa mallin luotettavuudesta ja ennusteen käyttökelpoisuudesta. (Jain 2001, 23-24.)

Yksittäinen ennustevirhe ei vielä kerro juuri mitään käytetystä aikasarjan mallintamismenetelmästä ja kiinnostavampaa onkin pyrkiä mittaamaan pidemmän aikavälin ennustevirhettä summaamalla yhteen ennustevirheitä. Seuraavassa kolme suuretta joiden avulla voidaan mitata ennusteen luotettavuutta.

Keskiarvoinen absoluuttinen poikkeama MAD (mean absolute deviation) painottaa kaikki virheet tasaisesti.

$$MAD = \frac{\sum_{t=1}^n |e_t|}{n}, \quad (23)$$

missä
 e = ennustevirhe jaksolla t ,
 n = jaksojen määrä.

Keskiarvoinen neliövirhe MSE (mean squared error) painottaa enemmän suuria virheitä kuin pieniä.

$$MSE = \frac{\sum_{t=1}^n e_t^2}{n}, \quad (24)$$

missä
 e = ennustevirhe jaksolla t ,
 n = jaksojen määrä.

MAD:n ja MSE:n tulokseen ei vaikuta se, onko virhe negatiivinen vai positiivinen. Ennuste on sitä tarkempi, mitä pienemmän arvon samalla aikavälillä MAD ja MSE saavat.

Keskiarvoinen absoluuttinen suhteellinen virhe MAPE (mean absolute percentage error) mittaa ennustevirheen suhteellista suuruutta todellisiin arvoihin.

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{x_t - y_t}{x_t} \right|, \quad (25)$$

missä
 x_t = todellinen aikasarjan arvo jaksolla t ,
 y_t = ennustettu aikasarjan arvo jaksolla t ja
 n = jaksojen määrä.

Aikasarjan malli on ennakkoluuloton ja tasapainoinen (ei sisällä systemaattista virhettä) jos ennustevirheet eivät autokorreloi ja niiden keskiarvo on lähellä nollaa.

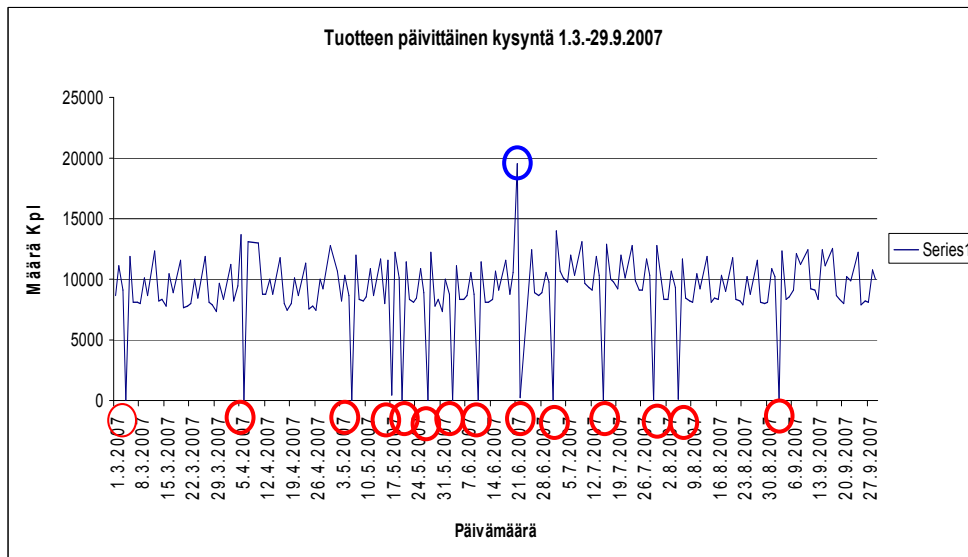
3. AIKASARJAN MALLINNUS JA ENNUSTEIDEN MUODOSTAMINEN

Aikasarjan mallintaminen alkaa aikasarjan visuaalisella tarkastelulla, jonka avulla pyritään havaitsemaan outlierit ja mahdolliset muut epätavalliset poikkeamat. Lisäksi tarkastellaan aikasarjan käyttäytymistä kuten sitä, onko havaittavissa jaksollisuutta. Visuaalisen tarkastelun jälkeen päätetään toimenpiteet aikasarjassa mahdollisesti olevin virheiden suhteen ja päätetään siitä, miten aikasarjaa ryhdytään neuroverkolla mallintamaan.

Tässä työssä, aikasarjan mallinnuksessa käytetään erään tuorepäivittäistavaran kysynnän aikasarjaa 1.3. – 29.9.2007. Aikasarja kuvaa tuotteen päivittäistä kysyntää koko Suomen alueella. Aikasarjan osaa 1.3. – 31.8.2007 käytetään neuroverkon opettamiseen ja opetetulla neuroverkolla ennustetaan kyseisen tuotteen päivittäistä kysyntää, syyskuussa 2007.

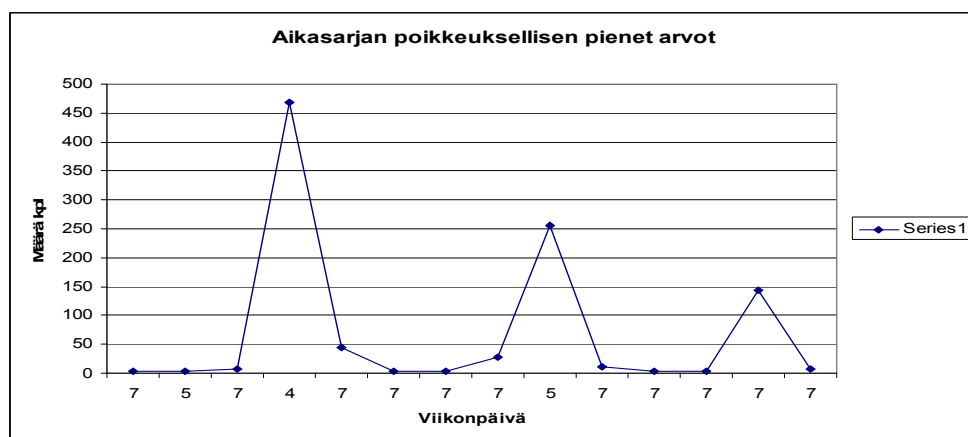
3.1 Visuaalinen tarkastelu ja esikäsittely

Aluksi tarkastellaan aikasarjaa visuaalisesti, aikasarjaa kuvaavan kaavion avulla ja pyritään löytämään aikasarjasta outlierit ja mahdolliset muut anomaliat. Aikasarjan kuvaajasta kuvassa 10 nähdään, että aikasarja sisältää poikkeuksellisen pieniä arvoja ja ainakin yhden poikkeuksellisen suuren arvon.



Kuva 13. Aikasarjan kuvaaja ja poikkeamat.

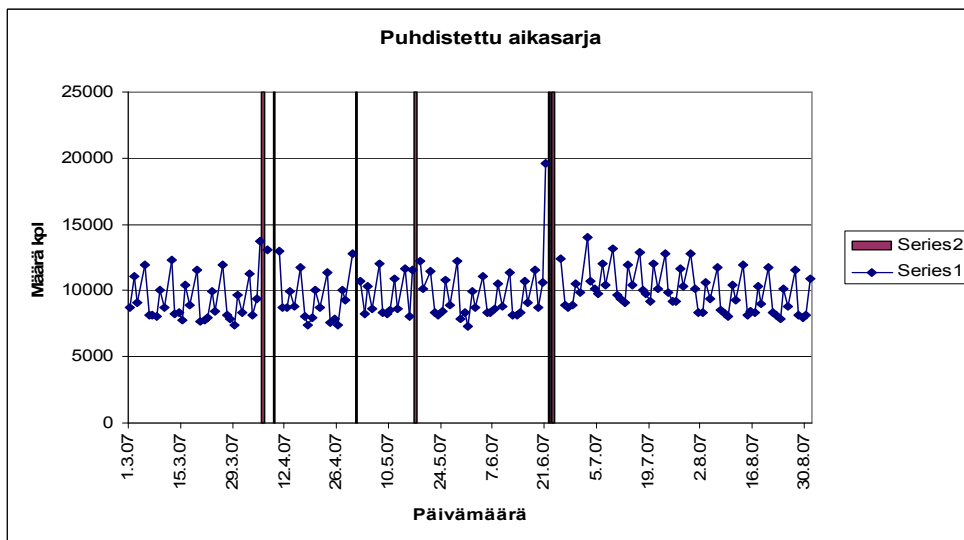
Tarkasteltaessa pieniä arvoja erikseen (kuva 11), huomataan että, suurin osa niistä on sunnuntaipäivien arvoja. Sunnuntaina ei kuitenkaan ole toimituksia, joten arvojen täytyy olla asiakkaiden erehdyksessä väärälle päivälle tekemiä tilauksia. Aikasarjasta ei voida päätellä kuuluvatko nämä pienet sunnuntaiset määrät edelliselle päivälle eli lauantaille, vai seuraavalle päivälle, eli maanantaille. Näin ollen jätämme nuo arvot kokonaan pois mallinnettavasta aikasarjasta.



Kuva 14. Aikasarjan pienet arvot.

Näiden epätavallisen pienten arvojen joukossa on kuitenkin myös kolme sellaista arvoa jotka eivät ole sunnuntaipäivien määriä. Nämä poikkeukset selittyvät myös väärille

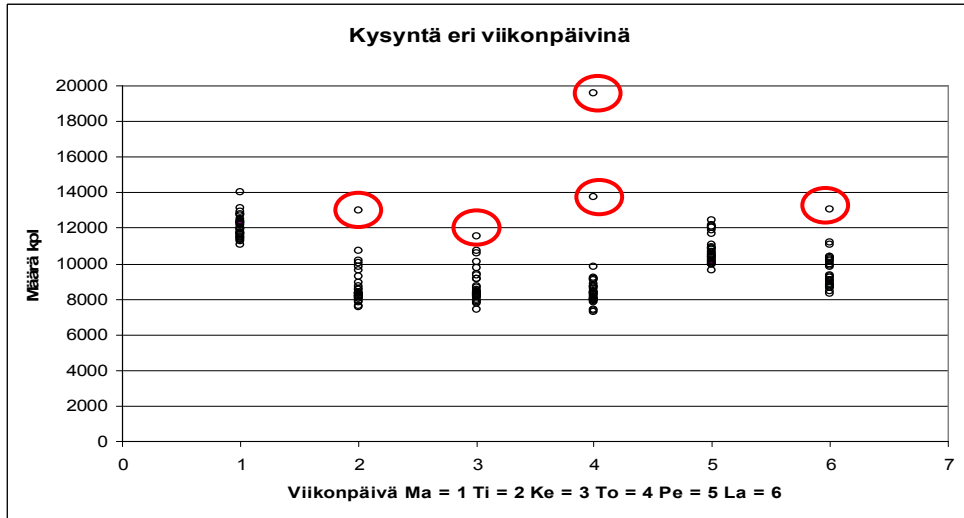
päiville tehtyinä tilauksina, koska kyseiset päivät ovat olleet vuonna 2007 juhlapyhä, jolloin ei siis ole myöskään ollut asiakastoimituksia. Kun aikasarjasta puhdistetaan pois nuo sunnuntaipäivien poikkeamat ja tarkastetaan että kaikilla muilla päivillä, kuin sunnuntaipäivillä, on jokin arvo, niin huomataan että juhlapyhät aiheuttavat aikasarjaan epäjatkuvuutta, sekä voimakkaita muutoksia aikasarjan arvoihin.



Kuva 15. Puhdistettu aikasarja (series1) ja epäjatkuvuus indikaattori (series2).

Nuo epäjatkuvuus kohdat täytyy "paikata" jollain tavalla, jotta neuroverkkoa voitaisiin opettaa ehyellä aikasarjalla. Tällöin tarkastellaan ensin aikasarjan käyttäytymistä päiväkohtaisesti ja päätetään tuo epäjatkuvuuskohdan paikkaustapa sen jälkeen.

Tarkasteltaessa tuotteen kysyntää eri viikonpäivinä, huomataan että kysyntä vaihtelee keskiarvon ympärillä suhteellisen vähän, lukuun ottamatta muutamia poikkeuksia. Kuvassa 13 on merkitty punaisella ympyrällä erikseen ne arvot jotka poikkeavat yli 25 % kyseisen päivän keskiarvosta.



Kuva 16. Tuotteen kysyntä eri viikonpäivinä ja ko. päivän keskiarvosta yli 25 % poikkeava arvo.

Tarkastellaan seuraavaksi mistä voi johtua yli 25 prosentin poikkeama saman viikonpäivän eri ajankohtina mitatuissa arvoissa. Taulukkoon 2 on koottu poikkeavien arvojen ajankohta ja selitys. Taulukosta nähdään, että kaikki poikkeavat arvot selittyvät juhlapyhien vaikutuksesta kysyntään. Kysyntä kasvaa juhlapyhien edellä kahdestakin syystä, ensinnä vähittäiskaupat ovat kiinni pyhinä, jolloin kysyntää ei siis voi olla ja myös siksi että pyhien aikaan on kulttuurillisista syistä tapana syödä enemmän ja paremmin.

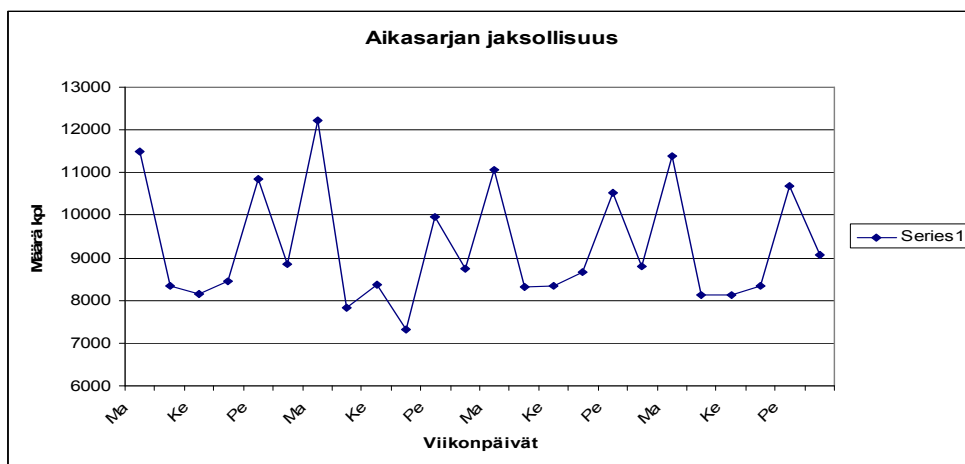
Taulukko 2. Suuren kysynnän päivät ja kysynnän selitykset.

Suuren kysynnän päivät 2007		
Päivämäärä	Määrä	Selitys
5.4.2007	13721	Pitkääperjantaita edeltävä torstai
7.4.2007	13092	Pääsiäispyhien välillä oleva arklauantai
10.4.2007	12991	Pääsiäisen jälkeinen ensimmäinen arkipäivä
16.5.2007	11560	Helatorstaita edeltävä keskiviikko
21.6.2007	19583	Juhannusta edeltävä torstai (perjantaina ei tuotantoa)

Kuten aikasarjan arvojen päivittäisessä tarkastelussa huomattiin, on arkipyhien aiheuttamia poikkeuksia lukuun ottamatta kysyntä kunakin arkipäivänä suhteellisen tasaista. Aikasarjan neuroverkkomallilla on tarkoitus ennustaa syyskuun kysyntää, eikä syyskuussa ole yhtäkään arkipyhään, täten pyritään arkipyhien vaikutus poistamaan

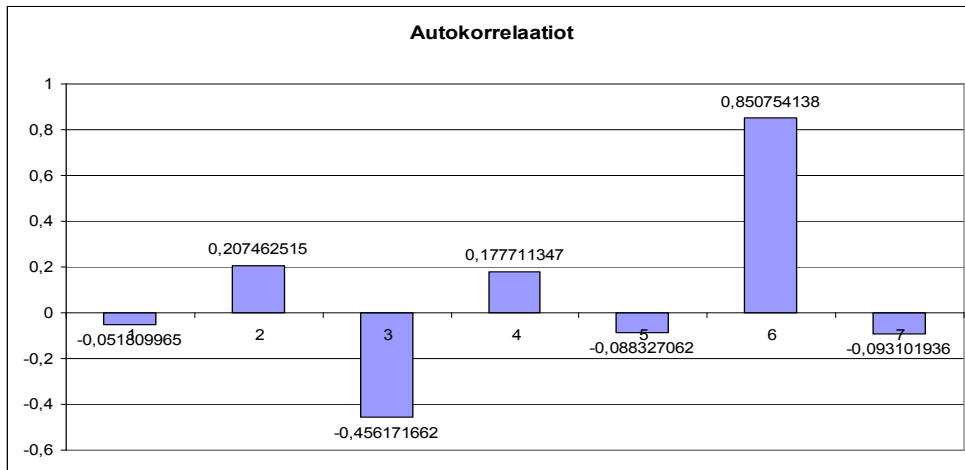
aikasarjasta. Vaikutus poistetaan korvaamalla arkipyhistä johtuvat puuttuvat arvot, sekä arkipyhän vaikutuksesta tulleet voimakkaasti keskiarvosta poikkeavat arvot, kunkin arvon edellisen ja seuraavan vastaavan päivän keskiarvolla. Tällöin mahdollinen pitkänajan trendi säilyy aikasarjassa.

Aikasarjaa tarkasteltaessa lyhyemmässä jaksossa, havaitaan aikasarjan jaksollisuus. Ajanjaksolla 21.5. – 16.6.2007, ei ole poikkeuksellisia arkipyhiä. Tällöin nähdään aikasarjan kuvaajasta (kuva 14) selvästi, että aikasarjassa on olemassa jaksollisuus, joka toistuu viikoittain. Jaksollisuus esiintyy siten, että maanantain kysytty määrä on aina suurempi kuin tiistain ja perjantain suurempi kuin lauantain, sekä aina saman viikonpäivän kysyntä on samansuuntainen, kuten kuvasta 12 nähtiin.



Kuva 17. Aikasarjan jaksollisuus.

Jaksollisuus näkyy myös kuvassa 15, kaavalla 2 lasketuissa autokorrelaatioissa eri viiveillä eli kuusi aikajaksoa taaksepäin on korrelaatio voimakkaimmillaan. Tästä havainnosta voidaan päätellä, että aikasarjaa neuroverkolla ennustettaessa, on parasta ottaa kuusi edeltävää arvoa mukaan ennusteen muodostamiseen, seitsemättä ei enää kannata ottaa, koska sen korrelaatio on hyvin pieni.



Kuva 18. Autokorrelaatiot eri viiveillä.

Kuten datan esi- ja jälkikäsittely kappaleessa todettiin, on neuroverkon helpompi oppia aikasarjanmalli ja vasteelle saadaan järkevä tulkinta, kun aikasarjan jokainen arvo skaalataan välille [0-1]. Aikasarjan arvot saadaan skaalattua välille [0-1] ja aikasarjan keskiarvoksi saadaan 0,5 seuraavalla kaavalla.

$$x' = \frac{x}{(\bar{x} / 0,5)} , \quad (26)$$

missä

x' = skaalattu aikasarjan arvo,

\bar{x} = aikasarjan keskiarvo,

x = aikasarjan alkuperäinen arvo,

0,5 = skaalari kerroin.

Vastaavasti neuroverkon vaste saadaan tulkittavaan muotoon kääntämällä edellinen kaava toisinpäin.

$$\hat{x} = \frac{\bar{x}}{0,5} x'' , \quad (27)$$

missä

\hat{x} = neuroverkon ennuste,

\bar{x} = aikasarjan keskiarvo,

x'' = neuroverkon antama vaste,
 0,5 = skaalari kerroin.

3.2 Neuroverkon konfigurointi

Neuroverkon konfiguroinnille ei ole olemassa mitään täsmällistä ohjetta, joten käytetään Koikkalaisen 1994 neuroverkkoraportissaan esittämää ohjenuoraa. Autokorrelaatioiden perusteella voidaan päätellä että aikasarja sisältää informaatiota edellisistä arvoista aina kuuden askeleen päähän.

Ohjenuora on, että särmien painokertoimien määrä ei saisi ylittää $3 \times$ syötevektorin dimension tuloa. Toisin sanoen 3×6 on maksimi määrä painokertoimia joka neuroverkossa sallitaan. Tähän painokertoimien määrään päästään kun neuroverkolla on 3 piiloneuronia.

Piiloneuroneiden määrän muodostuessa tässä aikasarjamallissa noin pieneksi, ei ole käytännössä järkevää testata neuroverkon toimintaa enää tuota pienemmällä piiloneuroneiden määrällä. Testataan kuitenkin voidaanko piiloneuroneiden määrän lisäämisestä saada lisäarvoa ennustuksen tuottamiseen.

Neuroverkon opettaminen tehdään siten, että ensin muodostetaan neuroverkko konfiguraatiolla $6 \times 3 \times 1$, eli kuusi syöteneuronia, kolme piiloneuronia ja yksi vastaneuroni. Neuroverkkoa opetetaan niin kauan kun opetusdatasta laskettava MAPE pienenee ja testiennusteista laskettu MAPE pienenee.

Opetus tapahtuu vaiheittain, niin että opetusdataa opetetaan neuroverkolle aina kymmenen opetuskierron ryhmissä. Saavutettaessa testiennuste, tuloksen minimipiste eli ennustevirhe kääntyy nousuun, tällöin lopetetaan opetus. Tämän jälkeen opetetaan neuroverkko sillä opetuskierron määrällä joka tuotti parhaimman tuloksen ja suoritetaan varsinaiset ennusteet. Lopuksi testataan vielä onko piilokerrosten

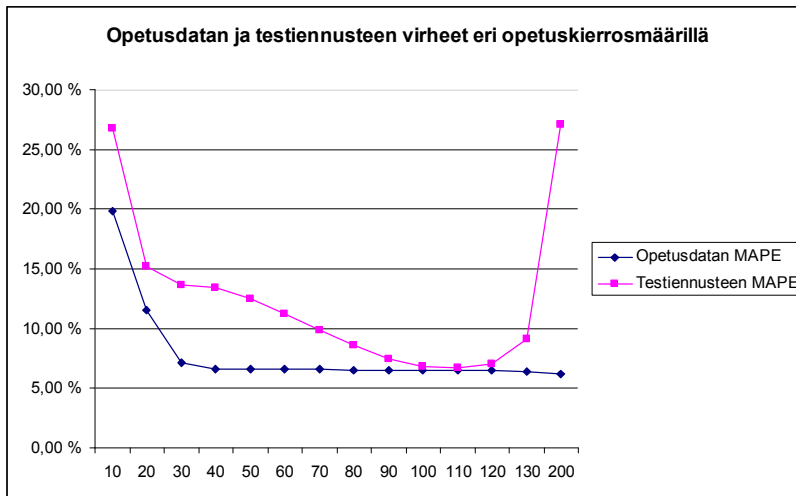
lisäämisestä hyötyä ja mikäli piilokerrosten lisäämisestä on hyötyä suoritetaan ennusteet myös muutetulla konfiguraatiolla.

3.3 Neuroverkon opettaminen opetus-datalla

Neuroverkkoa (6 X 3 X 1) opetettiin suunnitelman mukaan siten, että opetuskierroksia lisättiin kymmenen aina jokaisen koe-ennustuksen jälkeen. Taulukosta 3 ja kuvasta 16 nähdään että paras koe-ennustetulos saatiin 110 opetuskierroksen kohdalla, jonka jälkeen tulos alkaa heiketä. Opetettaessa neuroverkkoa 200 opetuskierrosta, se menetti yleistämiskykynsä totaalisesti, aivan kuten neuroverkkoteoria antoi olettaa.

Taulukko 3. Opetusdatan ja testiennusteen virheet eri opetuskierrosmäärillä kolmen piiloneuronin neuroverkossa.

Opetuskierrokset	Opetusdatan MAPE	Testiennusteen MAPE
10	19,82 %	26,79 %
20	11,51 %	15,17 %
30	7,16 %	13,60 %
40	6,65 %	13,39 %
50	6,63 %	12,43 %
60	6,61 %	11,21 %
70	6,58 %	9,84 %
80	6,54 %	8,60 %
90	6,52 %	7,43 %
100	6,51 %	6,81 %
110	6,50 %	6,68 %
120	6,49 %	7,07 %
130	6,42 %	9,12 %
200	6,16 %	27,06 %

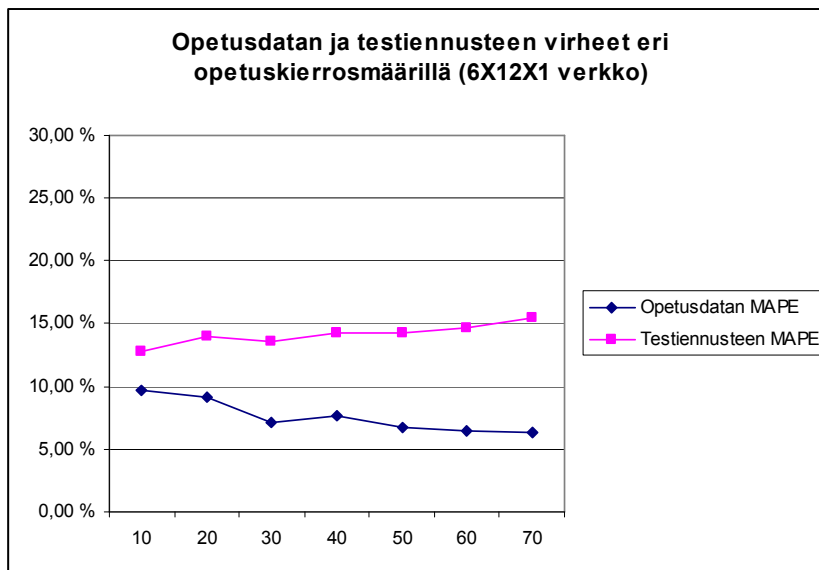


Kuva 19. Opetusdatan ja testiennusteen virheet eri opetuskierrosmäärillä kolmen piiloneuronin neuroverkossa

Seuraavaksi testattiin opetusta kahdentoista piiloneuronin neuroverkolla. Kävi hyvin nopeasti selväksi ettei sillä päästä likimainkaan samaan lopputulokseen kuin kolmen piiloneuronin verkolla, vaan verkko menetti yleistämiskykynsä jo opetuksen alkuvaiheessa kuten taulukosta 4, ja kuvasta 20 selviää.

Taulukko 4. Opetusdatan ja testiennusteen virheet eri opetuskierrosmäärillä kahdentoista piiloneuronin neuroverkossa.

Opetuskierrokset	Opetusdatan MAPE	Testiennusteen MAPE
10	9,72 %	12,80 %
20	9,19 %	14,04 %
30	7,16 %	13,60 %
40	7,68 %	14,31 %
50	6,78 %	14,21 %
60	6,49 %	14,73 %
70	6,29 %	15,47 %
80	6,07 %	16,48 %



Kuva 20. Opetusdatan ja testiennusteen virheet eri opetuskierrosmäärillä kahdentoista piiloneuronin neuroverkossa.

Näiden testien perusteella valittiin lopullisen ennusteen tuottamiseen neuroverkko jossa on 6 syöteneuronia, 3 piiloneuronia ja yksi vastaneuroni ja opetuskierroksia tehdään 110.

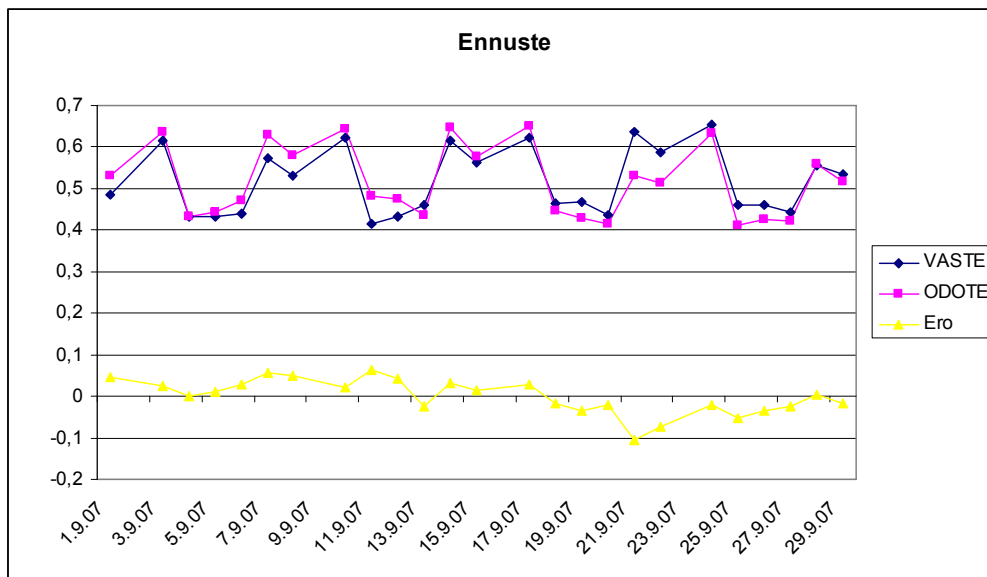
3.4 Ennusteiden muodostaminen

Edellisten neuroverkkojen opetuksen perusteella valittiin 6 X 3 X 1 eli kuusi syöteneuronia, kolme piiloneuronia ja yksi vastaneuroni. Valittua neuroverkkoa opetettiin 110 opetuskierrosta ja sillä ennustettiin syyskuun jokaiselle myyntipäivälle ennuste, siten että ennusteen pohjana käytetään aina kuuden edellisen päivän myyntimäärästä esikäsittellyt arvot. Nämä todelliset myyntimäärät eivät ole olleet aikaisemmin opetusdatassa eli ennuste on "aito" yhden askeleen ennuste aikasarjalle.

Taulukossa 5 on esitetty kaikille syyskuun myyntipäiville edellä mainitulla neuroverkolla tuotetut ennusteet.

Taulukko 5. Vaste, odote ja ero kullekin myyntipäivälle.

Päivämäärä	Viikonpäivä	VASTE	ODOTE	Ero
1.9.2007	Lauantai	0,486216241	0,53078	0,04456
3.9.2007	Maanantai	0,614813837	0,63808	0,02326
4.9.2007	Tiistai	0,432116299	0,43174	-0,0004
5.9.2007	Keskiviikko	0,432935571	0,444	0,01106
6.9.2007	Torstai	0,440802792	0,47038	0,02958
7.9.2007	Perjantai	0,571693454	0,62863	0,05693
8.9.2007	Lauantai	0,530340578	0,58064	0,0503
10.9.2007	Maanantai	0,623219601	0,6439	0,02068
11.9.2007	Tiistai	0,416205627	0,48134	0,06513
12.9.2007	Keskiviikko	0,43105866	0,47412	0,04306
13.9.2007	Torstai	0,460013964	0,43486	-0,0252
14.9.2007	Perjantai	0,614223886	0,64712	0,03289
15.9.2007	Lauantai	0,562721447	0,57565	0,01293
17.9.2007	Maanantai	0,622424028	0,65111	0,02869
18.9.2007	Tiistai	0,464260668	0,44789	-0,0164
19.9.2007	Keskiviikko	0,467117335	0,43086	-0,0363
20.9.2007	Torstai	0,435519648	0,41486	-0,0207
21.9.2007	Perjantai	0,636571367	0,53036	-0,1062
22.9.2007	Lauantai	0,586836587	0,51281	-0,074
24.9.2007	Maanantai	0,654654447	0,63268	-0,022
25.9.2007	Tiistai	0,462646308	0,41148	-0,0512
26.9.2007	Keskiviikko	0,461278147	0,42691	-0,0344
27.9.2007	Torstai	0,44437254	0,42156	-0,0228
28.9.2007	Perjantai	0,555898143	0,5594	0,0035
29.9.2007	Lauantai	0,536267237	0,51852	-0,0177

**Kuva 21.** Vasteen, odotteen ja eron kuvaajat kullekin myyntipäivälle.

4 ANALYYSI

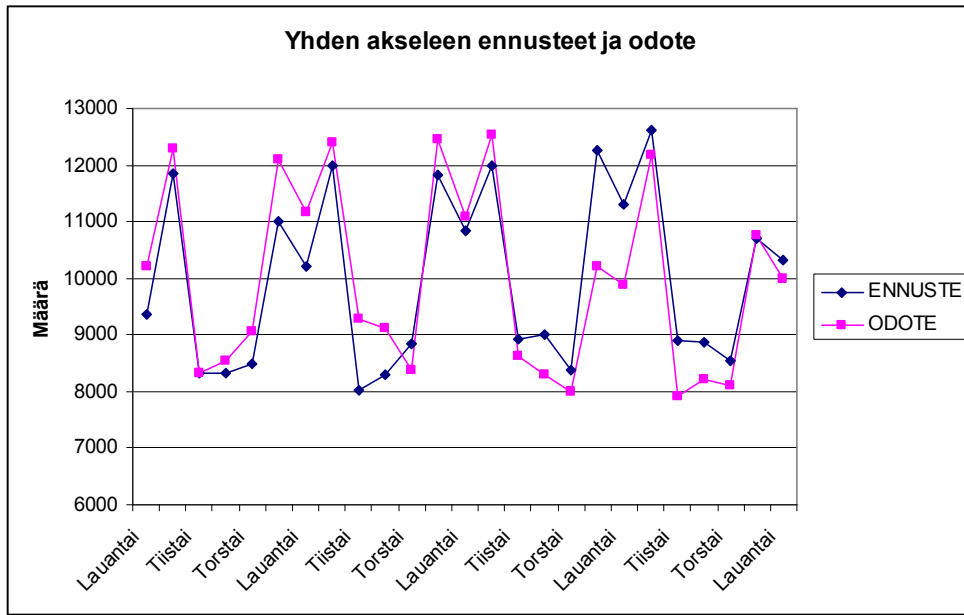
Muodostetun aikasarjan neuroverkkomallin ennustekyvyn mittaamiseksi lasketaan ennusteiden ja toteumien välisestä erosta tunnuslukuja, jotka kertovat mallin kyvykkyydestä tuottaa luotettavia ennusteita. *MAPE:lla* (kaava 25) selvitetään miten suuria ennustevirheet ovat suhteessa toteutuneisiin arvoihin. *MAD:n* (kaava 23) ja *MSE:n* (kaava 24) tulokseen ei vaikuta se, onko virhe negatiivinen vai positiivinen. Ennuste on sitä tarkempi, mitä pienemmän arvon samalla aikavälillä *MAD* ja *MSE* saavat. Aikasarjan malli on ennakkoluuloton ja tasapainoinen (ei sisällä systemaattista virhettä) jos ennustevirheet eivät korreloi ja niiden keskiarvo on lähellä nollaa.

4.1 Ennusteen tarkkuus ja aikasarjamallin ennustuskyky

Ennen laskelmia jälkikäsitellään neuroverkon antama ennuste (kaava 27), jolloin saadaan neuroverkon vasteesta kappalemääräinen ennuste. Taulukossa 6 on esitetty jokaisen myyntipäivän neuroverkon antama ennuste, sitä vastaava odote ja virheet, sekä absoluuttiset että suhteelliset. Kolmea ennustevirhettä lukuun ottamatta ennustevirheet ovat kohtuullisella tasolla. Ennuste seuraa odotetta suhteellisen hyvin kuten nähdään kuvasta 22.

Taulukko 6. Ennuste, odote ja virheet myyntipäivittäin.

Päivämäärä	Viikonpäivä	ENNUSTE	ODOTE	Virhe	ABS virhe	Suht. Virhe	Suht. ABS virhe
1.9.2007	Lauantai	9362	10220	-858	858	-8,40 %	8,40 %
3.9.2007	Maanantai	11838	12286	-448	448	-3,65 %	3,65 %
4.9.2007	Tiistai	8320	8313	7	7	0,09 %	0,09 %
5.9.2007	Keskiviikko	8336	8549	-213	213	-2,49 %	2,49 %
6.9.2007	Torstai	8488	9057	-569	569	-6,29 %	6,29 %
7.9.2007	Perjantai	11008	12104	-1096	1096	-9,06 %	9,06 %
8.9.2007	Lauantai	10212	11180	-968	968	-8,66 %	8,66 %
10.9.2007	Maanantai	12000	12398	-398	398	-3,21 %	3,21 %
11.9.2007	Tiistai	8014	9268	-1254	1254	-13,53 %	13,53 %
12.9.2007	Keskiviikko	8300	9129	-829	829	-9,08 %	9,08 %
13.9.2007	Torstai	8857	8373	484	484	5,79 %	5,79 %
14.9.2007	Perjantai	11827	12460	-633	633	-5,08 %	5,08 %
15.9.2007	Lauantai	10835	11084	-249	249	-2,25 %	2,25 %
17.9.2007	Maanantai	11985	12537	-552	552	-4,41 %	4,41 %
18.9.2007	Tiistai	8939	8624	315	315	3,65 %	3,65 %
19.9.2007	Keskiviikko	8994	8296	698	698	8,42 %	8,42 %
20.9.2007	Torstai	8386	7988	398	398	4,98 %	4,98 %
21.9.2007	Perjantai	12257	10212	2045	2045	20,03 %	20,03 %
22.9.2007	Lauantai	11299	9874	1425	1425	14,44 %	14,44 %
24.9.2007	Maanantai	12605	12182	423	423	3,47 %	3,47 %
25.9.2007	Tiistai	8908	7923	985	985	12,43 %	12,43 %
26.9.2007	Keskiviikko	8882	8220	662	662	8,05 %	8,05 %
27.9.2007	Torstai	8556	8117	439	439	5,41 %	5,41 %
28.9.2007	Perjantai	10704	10771	-67	67	-0,63 %	0,63 %
29.9.2007	Lauantai	10326	9984	342	342	3,42 %	3,42 %



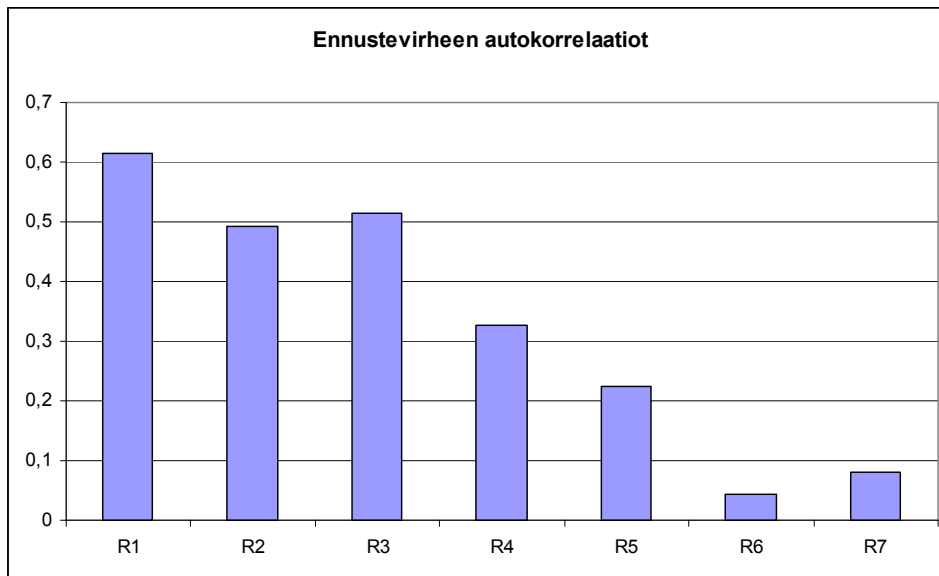
Kuva 22. Ennusteen ja odotteen kuvaajat.

Aikasarjamallin tunnuslukujen (taulukossa 7) perusteella neuroverkolla tuotettu aikasarjamalli tuottaa suhteellisen luotettavia ennusteita koko kuukauden jaksolla tarkasteltuna. MAPE joka kuvaa ennustevirheen absoluuttista suhteellista eroa ja MAD kuvaa absoluuttista keskiarvoista virhettä. Keskimäärin ennustevirhe on 654 tuotetta eli 7%. Ennustevirheiden keskiaro on hyvin lähellä nollaa, sama näkyy myös kuvassa 21, se kertoo neuroverkon tuottavan ennusteita odotetusti, sekä myös liian pieniä että liian suuria ennusteita tasaisesti. MSE arvolle ei ole vertailu kohtaa, joten sille ei saada tässä tapauksessa selkeää tulkintaa. MSE arvon avulla voidaan vertailla eri ennusteita, mitä pienempi luku sitä tarkempi ennuste.

Taulukko 7. Aikasarjamallin tunnusluvut.

MAPE	MAD	MSE	KA
7 %	654,426	629124	4

Virheiden autokorrelaatio on kuitenkin huomattava. Tämä kielii siitä, että neuroverkko ei saanut hyödynnettyä kaikkea aikasarjaan liittyvää informaatiota, vaan jotain jäi löytymättä.



Kuva 23. Ennustevirheiden autokorrelaatio aikaviiveillä 1 – 7 päivää.

Tarkasteltaessa aikasarjan tunnuslukuja tarkemmin, viikontäiväkohtaisesti (taulukko 8), paljastuu että ennustevirheiden keskiarvo ei säily lähellä nollaa viikontäiväkohtaisesti, vaan syntyy suurta hajontaa. MAPE liikkuu samoin välillä 3,68% – 7,48%.

Taulukko 8. Aikasarjamallin tunnusluvut viikontäiväkohtaisesti.

Viikontäivä	MAPE	MAD	KA
Maanantai	3,68 %	455	-244
Tiistai	7,43 %	640	13
Keskiviikko	7,01 %	601	79
Torstai	5,62 %	473	188
Perjantai	8,70 %	960	62
Lauantai	7,43 %	769	-62

5. YHTEENVETO

Paras MLP-neuroverkko kullekin aikasarjalle löytyy ainoastaan testaamalla, mitään yksiselitteistä ohjetta ei ole. Tuotetun mallin kyky tuottaa ennusteita riippuu mallin tekijän ammattitaidosta määritellä neuroverkon rakenne sekä tietysti aikasarjasta itsestään.

Muodostettu aikasarjamalli MLP-neuroverkolla antoi hyviä ennustetuloksia, jopa koko neljän viikon jaksolla. Muutamia suurempia virheitä kuitenkin oli ja tutkittavaksi vielä jää, olisiko ennustetulos vielä parempi jos käytettäisiin myös saman tuoteryhmän muiden nimikkeiden kysyntää aikasarjan mallissa? Voitaisiinko tuollaisella mallilla tutkia jopa muiden tuoteryhmän tuotteiden kysynnän muutosten vaikutusta kyseisen tuotteen kysyntään? Olisiko tulos ollut vielä tarkempi jos jokaisesta viikonpäivästä olisi tehty oma neuroverkkomalli?

Tieteellisestä näkökulmasta tarkasteltuna, on neuroverkkojen testaaminen ongelmallista siksi, että algoritmissa alustetaan särmien painokertoimet satunnaisluvuilla. Tästä sekä desimaalien pienistä pyöristysvirheistä ja opetusalgoritmista johtuen ei juuri koskaan päädytä täsmälleen samaan lopputulokseen, ei ennusteen, eikä neuroverkon painokertoimienkaan suhteen.

Tässä työssä kehitettyä VBA-Excel neuroverkkoa voidaan kuitenkin käytännön tasollakin käyttää, pienessä mittakaavassa. Ohjelman kehittäminen antoi myös eväitä ymmärtää neuroverkkojen toimintaa ja niiden käytön rajoitteita.

Neuroverkkojen idea, aivojen "matkiminen", on hyvin mielenkiintoinen ja mukaansatempaava aihe, joka on vuosien saatossa kehittynyt. Mielenkiintoista onkin se, löydetäänkö ja voidaanko käytännössä toteuttaa, aivoja todella jäljittelevä malli kaikkine kompleksisuuksineen. Odotukset ovat mielestäni olleet tähän saakka liian kunnianhimoisia. Miten olisikaan voitu odottaa, että tietokone voisi vielä nykyisellä tehokkuudella ja rakenteella, päästä aivojen tasolle?

LÄHDELUETTELO

Haykin, Simon (1999). Neural Networks: A Comprehensive Foundation. 2. painos. New Jersey: Prentice Hall.

Buffa, Elwood & Sarin, Rakesh (1987). Modern Production / Operations Management. 8th edition.

Granger, C.W.J. (1989). Forecasting in Business and Economics. 2nd edition. San Diego, London: Academic Press, Inc.

Heikki Huttunen (2005). Signaalinkäsittelyn menetelmät. Tampereen teknillinen yliopisto 2005.

Ilkka Kokkarinen (2003). Tekoäly laskettavuus ja logiikka Talentum.

Ilkka Kokkarinen & Kirsti Ala-Mutka (2002). Tietorakenteet ja algoritmit Satku Kauppakaari

Jacek M. Zurada (1992). Introduction to artificial neural systems, West Publishing Company

James A. Freeman & Davis M. Skapura (1992). Neural Networks algorithms, Applications and Programins Techniques Addison- Wesley Publishing inc.

Koikkalainen, Pasi (1994). Neurolaskennan mahdollisuudet. Tekesin julkaisut 43/94.

Masters, Timothy (1993). Practical Neural Network Recipes in C++. 1. painos. San Francisco: Morgan Kaufmann.

LIITE 1

Excel-sovelluksen VBA- koodi

```
'Jyrki Kankaanpää m72855
'Vaasan Yliopisto
'MPL neuroverkko
'Algoritmi toteutettu kirjan Introduction to artificial neural systems
'Zurada, Jacek M 1992 s.186 -191 mukaan

'Muuttujien määrittelyt
Option Explicit
Dim input_node           'syöte neuronien määrä
Dim hide_node            'piilo neuronien
määrä
Dim out_node             'vaste neuronien määrä
Dim input_hide_funktio   'käytettävä akvaatio funktio
Dim kierrokset           'määrätyt opetuskierrokset
Dim tarkkuus             'määrätty oppimistarkkuus
Dim Counter              'opetuskierros laskuri
Dim inputtxt             'soluntunnistekenttä
Dim outtxt               'soluntunnistekenttä
Dim odotetxt             'soluntunnistekenttä
Dim odotetut_vasteet_virheet_out() 'Odotettujen vasteiden ja korjaustermien
matriisi
Dim odotetut_vasteet_virheet_hide() 'Odotettujen vasteiden ja korjaustermien
matriisi
Dim input_vector()       'Syöte vektori
Dim verkon_virhe         'verkon vasteen virhe suhteessa
odotteeseen
Dim bias_I               'Input_hide bias
Dim bias_II              'Hide_out bias
Dim moment               'edellisen korjauksen vaikutus
momentti

Dim rivi                  'Opetus tai tutkittavan datan rivi
Dim oppimisnopeus        'Kerroin jolla korjausta
painotetaan
Dim nykyinen_paino       'muutettava painokerroin
Dim i, j, k, m, n, l, o, p
Dim hide_summa As Double
Dim out_summa As Double
Dim alfa                  'inputtien ja painojen summaus
Dim virhe_summa_hide     'hideinputtien ja painojen summaus
                        'aktivaatio funktion jyrkkyys
                        'Virhe_out*hide painot

Sub Luo_verkko()

    inputtxt = "INPUT"
    outtxt = "OUT"
    odotetxt = "ODOTE"

    'Montako input nodea
    input_node = Sheet3.Cells(3, 3).Value
    'Montako Hide nodet
    hide_node = Sheet3.Cells(3, 4).Value
    'Montako Out node
    out_node = Sheet3.Cells(3, 5).Value

    'tyhjäetään vanhat nodet
    For i = 10 To 200
        For j = 3 To 10
            Sheet3.Cells(i, j).Value = Null
        Next j
    Next i
End Sub
```

```

Next i

'Luodaan input nodet
Sheet3.Cells(10, 3).Value = "INPUT"
Sheet3.Cells(10, 4).Value = "ARVO"
For i = 1 To input_node
    For j = 1 To 2
        Sheet3.Cells(10 + i, (3 + j - 1)).Value = i
    Next j
Next i

'Luodaan hide nodet
Sheet3.Cells(10, 5).Value = "HIDE"
Sheet3.Cells(10, 6).Value = "ARVO"
For m = 1 To hide_node
    For n = 1 To 2
        Sheet3.Cells(10 + m, (5 + n - 1)).Value = m
    Next n
Next m

'Luodaan out nodet
Sheet3.Cells(10, 7).Value = "OUT"
Sheet3.Cells(10, 8).Value = "ARVO"
For p = 1 To out_node
    For o = 1 To 2
        Sheet3.Cells(10 + p, (7 + o - 1)).Value = p
    Next o
Next p

'Luodaan odotteet nodet
Sheet3.Cells(10, 9).Value = "ODOTE"
Sheet3.Cells(10, 10).Value = "ARVO"
For p = 1 To out_node
    For o = 1 To 2
        Sheet3.Cells(10 + p, (9 + o - 1)).Value = p
    Next o
Next p

'Poistetaan vanhat otsikot
For i = 1 To 200
    Sheet1.Cells(1, i).Value = Null
Next i
For i = 1 To 200
    Sheet2.Cells(1, i).Value = Null
Next i

'kijoitetaan soluotsikot data sivulle
Sheet1.Cells(1, 1).Value = "TUNNISTE"
For i = 1 To input_node
    Sheet1.Cells(1, i + 1).Value = inputtxt & i
Next i
For i = input_node + 1 To (input_node + out_node)
    Sheet1.Cells(1, i + 1).Value = outtxt & (i - input_node)
Next i

'kijoitetaan soluotsikot opetus data sivulle
Sheet2.Cells(1, 1).Value = "TUNNISTE"
For i = 1 To input_node
    Sheet2.Cells(1, i + 1).Value = inputtxt & i
Next i
For i = input_node + 1 To (input_node + out_node)
    Sheet2.Cells(1, i + 1).Value = odotetxt & (i - input_node)
Next i
For i = (input_node + out_node + 1) To (input_node + out_node + out_node)
    Sheet2.Cells(1, i + 1).Value = outtxt & (i - input_node - out_node)
Next i

MsgBox (input_node & " X " & hide_node & " X " & out_node & "verkko luotu")
Call alusta_painot
End Sub
Sub alusta_painot()

'tyhjätään vanhat

```

```

For i = 1 To 200
    For j = 1 To 200
        Sheet4.Cells(i, j).Value = Null
    Next j
Next i

'alustetaan pienillä satunnaisluvuilla input_hide väliset painot
'lisätään yksi bias input varten
For i = 1 To input_node + 1
    For j = 1 To hide_node
        Sheet4.Cells(i, j).Value = (1.5 * Rnd) - 0.5
    Next j
Next i

'tyhjätään vanhat
For i = 1 To 200
    For j = 1 To 200
        Sheet5.Cells(i, j).Value = Null
    Next j
Next i

'alustetaan pienillä satunnaisluvuilla hide_out väliset painot
'lisätään yksi bias hide varten
For i = 1 To hide_node + 1
    For j = 1 To out_node
        Sheet5.Cells(i, j).Value = (1.5 * Rnd) - 0.5
    Next j
Next i

MsgBox (input_node & " X " & hide_node & " matriisi ja " & hide_node & " X " &
out_node & " matriisi luotu")

End Sub

Sub opeta_data()

    'haetaan käytettävä akvaatio funktion koodi
    input_hide_funktio = Sheet3.Cells(2, 9).Value

    'Haetaan biaskset
    bias_I = Sheet3.Cells(4, 7).Value
    bias_II = Sheet3.Cells(5, 7).Value
    'Input_hide bias
    'Hide_out bias

    'lisätään tila biakselle
    'Montako input nodea
    input_node = Sheet3.Cells(3, 3).Value + 1
    'Montako Hide nodet
    hide_node = Sheet3.Cells(3, 4).Value + 1
    'Montako Out node
    out_node = Sheet3.Cells(3, 5).Value
    'Määritellään taulukkomuuttujat uudelleen vastaavan verkon kokoideksi
    ReDim odotetut_vasteet_virheet_out(out_node, 3)
    ReDim odotetut_vasteet_virheet_hide(hide_node, 3)
    ReDim input_vector(input_node, 1)

    'Nollataan laskin
    Counter = 0

    'asetetaan toivottu verkon tarkkuus
    tarkkuus = Sheet3.Cells(5, 5).Value
    'Asetetaan opetus kierrokset
    kierrokset = Sheet3.Cells(4, 5).Value
    'asetetaan korjauksen nopeus
    oppimisnopeus = Sheet3.Cells(6, 5).Value
    'Millä rivillä on ensimmäinen opetus tapaus
    rivi = 2

    'Silmukka kiertää niin kauan kunnes opetus kierrokset täyttyy tai
    'verkon virhe on pienempää kuin asetettu
    While Counter < kierrokset
        'Silmukka kiertää niin kauan kun löytyy opetus rivejä
        While (Sheet2.Cells(rivi, 1).Value <> 0)

```

```

'haetaan input arvot
input_vector(input_node, 1) = bias_I
For i = 1 To input_node - 1
    'Haetaan opetus data sivulta
    input_vector(i, 1) = Sheet2.Cells(rivi, i + 1).Value
    'Esitetään verkko sivulla
    Sheet3.Cells(10 + i, 4).Value = input_vector(i, 1)
Next i
'MsgBox ("Input " & input_vector(input_node, 1))
'haetaan odotteet arvot näytettäväksi verkko sivulla
For i = 1 To out_node
    odotetut_vasteet_virheet_out(i, 1) = Sheet2.Cells(rivi, (input_node + i)).Value
    Sheet3.Cells(10 + i, 10).Value = Sheet2.Cells(rivi, (input_node + i)).Value
Next i

'lasketaan hide neuronien vasteet
hide_summa = 0
odotetut_vasteet_virheet_hide(hide_node, 2) = bias_II
For k = 1 To hide_node - 1
    For i = 1 To input_node
        hide_summa = hide_summa + _
        input_vector(i, 1) * Sheet4.Cells(i, k).Value
        'Sheet3.Cells(10 + i, 4).Value * Sheet4.Cells(i, k).Value
    Next i
    'Valitaan aktivointi funktio
    Select Case input_hide_funktio
        Case 1
            odotetut_vasteet_virheet_hide(k, 2) = _
            bipo_sigmoid(hide_summa)
            'näytetään käyttäjälle
            Sheet3.Cells(10 + k, 6).Value = _
            odotetut_vasteet_virheet_hide(k, 2)
        Case 2
            odotetut_vasteet_virheet_hide(k, 2) = _
            binaari_sigmoid(hide_summa)
            'näytetään käyttäjälle
            Sheet3.Cells(10 + k, 6).Value = _
            odotetut_vasteet_virheet_hide(k, 2)
        Case Else
            MsgBox ("Virhe ohjelma pysähtyy")
            Exit Sub
    End Select

Next k
'MsgBox ("Hide " & odotetut_vasteet_virheet_hide(hide_node, 2))
'testi
'i = 0
'For i = 1 To input_node
'    Sheet3.Cells(13 + i, 1).Value = odotetut_vasteet_virheet_hide(i, 2)
'Next i

'lasketaan out neuronien vasteet ja lasketaan virheet
out_summa = 0
verkon_virhe = 0
For k = 1 To out_node
    For i = 1 To hide_node
        out_summa = out_summa + _
        odotetut_vasteet_virheet_hide(k, 2) * Sheet5.Cells(i, k).Value
        'Sheet3.Cells(10 + i, 6).Value * Sheet5.Cells(i, k).Value
    Next i
    'Valitaan aktivointi funktio
    Select Case input_hide_funktio
        Case 1
            odotetut_vasteet_virheet_out(k, 2) = _
            bipo_sigmoid(out_summa)
            'Näytetään käyttäjälle vaste
            Sheet3.Cells(10 + k, 8).Value = _
            odotetut_vasteet_virheet_out(k, 2)
            'Laitetaan opetus data sivulle virheen laskentaa varten vaste
            Sheet2.Cells(rivi, input_node + out_node + k).Value = _

```

```

        odotetut_vasteet_virheet_out(k, 2)
        'virheen summaus
        verkon_virhe = verkon_virhe + _
        0.5 * ((Abs(odotetut_vasteet_virheet_out(k, 2) - _
        odotetut_vasteet_virheet_out(k, 1))) ^ 2)
    Case 2
        odotetut_vasteet_virheet_out(k, 2) = _
        binaari_sigmoid(out_summa)
        Sheet3.Cells(10 + k, 8).Value = _
        odotetut_vasteet_virheet_out(k, 2)
        Sheet2.Cells(rivi, input_node + out_node + k).Value = _
        odotetut_vasteet_virheet_out(k, 2)

        'virheen summaus
        verkon_virhe = verkon_virhe + _
        0.5 * ((Abs(odotetut_vasteet_virheet_out(k, 2) - _
        odotetut_vasteet_virheet_out(k, 1))) ^ 2)

    Case Else
        MsgBox ("Virhe ohjelma pysähtyy")
        Exit Sub
    End Select
Next k

'Opetus vaihe alkaa

'lasketaan korjaustermit out
' valitaan kummalla tavalla korjaustermit lasketaan riippuen funktiosta
If input_hide_funktio = 2 Then

    '(odotettu vaste - verkonvaste) * (1- verkonvaste) * verkonvaste)
    For k = 1 To out_node
        odotetut_vasteet_virheet_out(k, 3) = _
        (odotetut_vasteet_virheet_out(k, 1) - odotetut_vasteet_virheet_out(k,
2)) _
        * (1 - odotetut_vasteet_virheet_out(k, 2)) *
        odotetut_vasteet_virheet_out(k, 2)

    Next k
    'lasketaan korjaustermit hide
    'verkonvaste * (1- verkonvaste) * Summa(virhetermi_out * paino_hide_out)
    For j = 1 To hide_node
        For k = 1 To out_node
            virhe_summa_hide = Sheet5.Cells(j, k).Value *
            odotetut_vasteet_virheet_out(k, 3)
        Next k
        odotetut_vasteet_virheet_hide(j, 3) = _
        odotetut_vasteet_virheet_hide(j, 2) * _
        (1 - odotetut_vasteet_virheet_hide(j, 2)) * _
        virhe_summa_hide
    Next j
Else
    '0,5 * (odotettu vaste - verkonvaste) * (1- verkonvaste^2)
    For k = 1 To out_node
        odotetut_vasteet_virheet_out(k, 3) = _
        0.5 * (odotetut_vasteet_virheet_out(k, 1) -
        odotetut_vasteet_virheet_out(k, 2)) _
        * (1 - odotetut_vasteet_virheet_out(k, 2) ^ 2)

    Next k
    'lasketaan korjaustermit hide
    '0.5 * (1- verkonvaste^2) * Summa(virhetermi_out * paino_hide_out)
    For j = 1 To hide_node
        For k = 1 To out_node
            virhe_summa_hide = Sheet5.Cells(j, k).Value *
            odotetut_vasteet_virheet_out(k, 3)
        Next k
        odotetut_vasteet_virheet_hide(j, 3) = _
        0.5 * (1 - odotetut_vasteet_virheet_hide(j, 2) ^ 2) * _
        virhe_summa_hide
    Next j
End If

```

```

'korjataan painot
'out taso
For i = 1 To out_node
    For j = 1 To hide_node
        nykyinen_paino = Sheet5.Cells(j, i).Value
        Sheet5.Cells(j, i).Value = nykyinen_paino + (oppimisnopeus * _
            odotetut_vasteet_virheet_out(i, 3) * odotetut_vasteet_virheet_out(i, 1))
    Next j
Next i
'hide taso
For i = 1 To hide_node - 1
    For j = 1 To input_node
        nykyinen_paino = Sheet4.Cells(j, i).Value
        Sheet4.Cells(j, i).Value = nykyinen_paino + (oppimisnopeus * _
            odotetut_vasteet_virheet_hide(i, 3) * input_vector(j, 1))
    Next j
Next i
'tämänhetkinen opetus rivi
Sheet3.Cells(8, 5).Value = (rivi - 1)
'vaihdetaan riviä
rivi = rivi + 1

Wend
'Lisätään kierros laskuriin ja esitetään se
Counter = Counter + 1
Sheet3.Cells(7, 5).Value = Counter
rivi = 2
'Esitetään verkon virhe
Sheet3.Cells(9, 5).Value = verkon_virhe
'jos virhe pienempää kuin haluttu tarkkuus keskeytetään opetus
If verkon_virhe < tarkkuus Then
    Counter = Sheet3.Cells(4, 5).Value
End If

Wend

End Sub
Sub tutki_data()

    rivi = 2
    'haetaan käytettävä akvaatio funktion koodi
    input_hide_funktio = Sheet3.Cells(2, 9).Value

    'Haetaan biaskset
    bias_I = Sheet3.Cells(4, 7).Value
    bias_II = Sheet3.Cells(5, 7).Value
    'Input_hide bias
    'Hide_out bias

    'lisätään tila biakselle
    'Montako input nodea
    input_node = Sheet3.Cells(3, 3).Value + 1
    'Montako Hide nodet
    hide_node = Sheet3.Cells(3, 4).Value + 1
    'Montako Out node
    out_node = Sheet3.Cells(3, 5).Value
    '
    'silmutta pyörii niin kauan kun on tutkittavia rivejä
    While (Sheet1.Cells(rivi, 1).Value <> 0)
        input_vector(input_node, 1) = bias_I
        For i = 1 To input_node - 1
            'Haetaan data sivulta
            input_vector(i, 1) = Sheet1.Cells(rivi, i + 1).Value
            'Esitetään verkko sivulla
            Sheet3.Cells(10 + i, 4).Value = input_vector(i, 1)
        Next i

        'lasketaan hide neuronien vasteet
        hide_summa = 0
        odotetut_vasteet_virheet_hide(hide_node, 2) = bias_II
        For k = 1 To hide_node - 1
            For i = 1 To input_node

```



```

        hide_summa = hide_summa + _
        input_vector(i, 1) * Sheet4.Cells(i, k).Value
        'Sheet3.Cells(10 + i, 4).Value * Sheet4.Cells(i, k).Value
    Next i
    'Valitaan aktivointi funktio
    Select Case input_hide_funktio
    Case 1
        odotetut_vasteet_virheet_hide(k, 2) = _
        bipo_sigmoid(hide_summa)
        'näytetään käyttäjälle
        Sheet3.Cells(10 + k, 6).Value = _
        odotetut_vasteet_virheet_hide(k, 2)
    Case 2
        odotetut_vasteet_virheet_hide(k, 2) = _
        binaari_sigmoid(hide_summa)
        'näytetään käyttäjälle
        Sheet3.Cells(10 + k, 6).Value = _
        odotetut_vasteet_virheet_hide(k, 2)
    Case Else
        MsgBox ("Virhe ohjelma pysähtyy")
        Exit Sub
    End Select

Next k

'laskeaan out neuronien vasteet
out_summa = 0
For k = 1 To out_node
    For i = 1 To hide_node
        out_summa = out_summa + _
        odotetut_vasteet_virheet_hide(k, 2) * Sheet5.Cells(i, k).Value
        'Sheet3.Cells(10 + i, 6).Value * Sheet5.Cells(i, k).Value
    Next i
    'Valitaan aktivointi funktio
    Select Case input_hide_funktio
    Case 1
        odotetut_vasteet_virheet_out(k, 2) = _
        bipo_sigmoid(out_summa)
        'Näytetään käyttäjälle vaste
        Sheet3.Cells(10 + k, 8).Value = _
        odotetut_vasteet_virheet_out(k, 2)
        Sheet1.Cells(rivi, input_node + out_node).Value = _
        odotetut_vasteet_virheet_out(k, 2)
    Case 2
        odotetut_vasteet_virheet_out(k, 2) = _
        binaari_sigmoid(out_summa)
        Sheet3.Cells(10 + k, 8).Value = _
        odotetut_vasteet_virheet_out(k, 2)
        Sheet1.Cells(rivi, input_node + out_node).Value = _
        odotetut_vasteet_virheet_out(k, 2)

    Case Else
        MsgBox ("Virhe ohjelma pysähtyy")
        Exit Sub
    End Select
Next k

rivi = rivi + 1
Wend
End Sub
Function bipo_sigmoid(summa As Double) As Double
    alfa = Sheet3.Cells(6, 7).Value
    bipo_sigmoid = (1 - Exp(1) ^ (-alfa * summa)) _
    / (1 + Exp(1) ^ (-alfa * summa))
End Function
Function binaari_sigmoid(summa As Double) As Double
    summa = Round(summa, 5)
    alfa = Sheet3.Cells(6, 7).Value

```

```
binaari_sigmoid = Round((1 / (1 + (Exp(1) ^ (-alfa * summa)))), 5)
End Function
```

LIITE 2

Työssä käytetty "raaka" aikasarjadata

Päivämäärä	Määrä	Päivämäärä	Määrä	Päivämäärä	Määrä	Päivämäärä	Määrä
20070301	8711	20070508	8304	20070710	9612	20070911	9268
20070302	11094	20070509	8264	20070711	9366	20070912	9129
20070303	9077	20070510	8569	20070712	9121	20070913	2764
20070304	4	20070511	10876	20070713	11904	20070914	12460
20070305	11940	20070512	8642	20070714	10372	20070915	11084
20070306	8151	20070514	11676	20070715	4	20070917	12537
20070307	8097	20070515	8008	20070716	12925	20070918	8624
20070308	8054	20070516	11560	20070717	10036	20070919	8296
20070309	10075	20070517	468	20070718	9778	20070920	7988
20070310	8668	20070518	12183	20070719	9216	20070921	10212
20070312	12316	20070519	10122	20070720	12030	20070922	9874
20070313	8277	20070520	44	20070721	10150	20070924	12182
20070314	8304	20070521	11483	20070723	12741	20070925	7923
20070315	7805	20070522	8344	20070724	9840	20070926	8220
20070316	10427	20070523	8157	20070725	9149	20070927	8117
20070317	8880	20070524	8439	20070726	9147	20070928	10771
20070319	11508	20070525	10840	20070727	11668	20070929	9984
20070320	7643	20070526	8865	20070728	10296		
20070321	7757	20070527	4	20070729	4		
20070322	7977	20070528	12215	20070730	12814		
20070323	9975	20070529	7819	20070731	10143		
20070324	8452	20070530	8356	20070801	8359		
20070326	11912	20070531	7314	20070802	8370		
20070327	8140	20070601	9957	20070803	10620		
20070328	7889	20070602	8755	20070804	9368		
20070329	7350	20070603	4	20070805	144		
20070330	9654	20070604	11071	20070806	11696		
20070331	8340	20070605	8326	20070807	8496		
20070402	11238	20070606	8338	20070808	8237		
20070403	8172	20070607	8657	20070809	8075		
20070404	9407	20070608	10520	20070810	10394		
20070405	13721	20070609	8793	20070811	9252		
20070406	4	20070610	28	20070813	11928		
20070407	13092	20070611	11397	20070814	8148		
20070410	12991	20070612	8121	20070815	8462		
20070411	8740	20070613	8138	20070816	8290		
20070412	8735	20070614	8342	20070817	10344		
20070413	9979	20070615	10679	20070818	9032		

20070414	8783	20070616	9080	20070820	11749
20070416	11773	20070618	11594	20070821	8376
20070417	8042	20070619	8724	20070822	8178
20070418	7407	20070620	10573	20070823	7900
20070419	7979	20070621	19583	20070824	10172
20070420	10081	20070622	256	20070825	8770
20070421	8719	20070625	12402	20070827	11537
20070423	11335	20070626	8940	20070828	8153
20070424	7570	20070627	8668	20070829	7990
20070425	7829	20070628	8872	20070830	8164
20070426	7403	20070629	10521	20070831	10900
20070427	10020	20070630	9815	20070901	10220
20070428	9234	20070701	12	20070902	8
20070430	12758	20070702	13990	20070903	12286
20070502	10698	20070703	10688	20070904	8313
20070503	8230	20070704	10096	20070905	8549
20070504	10323	20070705	9801	20070906	9057
20070505	8644	20070706	12030	20070907	12104
20070506	8	20070707	10388	20070908	11180
20070507	12033	20070709	13159	20070910	12398

LIITE 3

Neuroverkoissa käytetty "putsattu" aikasarja

Määrä	Järjestelmän päivämäärä	PVM	ViikoPäivä	Muunnet	Keskiarvo	Munnettujen keskiaro
8711	20070301	1.3.2007	4	0,45240955	9627,34	0,5
11094	20070302	2.3.2007	5	0,57617168		
9077	20070303	3.3.2007	6	0,47141792		
11940	20070305	5.3.2007	1	0,62010906		
8151	20070306	6.3.2007	2	0,42332571		
8097	20070307	7.3.2007	3	0,42052119		
8054	20070308	8.3.2007	4	0,41828797		
10075	20070309	9.3.2007	5	0,52324948		
8668	20070310	10.3.2007	6	0,45017633		
12316	20070312	12.3.2007	1	0,63963678		
8277	20070313	13.3.2007	2	0,42986957		
8304	20070314	14.3.2007	3	0,43127183		
7805	20070315	15.3.2007	4	0,40535605		
10427	20070316	16.3.2007	5	0,54153075		
8880	20070317	17.3.2007	6	0,46118664		
11508	20070319	19.3.2007	1	0,59767295		
7643	20070320	20.3.2007	2	0,39694251		
7757	20070321	21.3.2007	3	0,40286315		
7977	20070322	22.3.2007	4	0,41428894		
9975	20070323	23.3.2007	5	0,51805593		
8452	20070324	24.3.2007	6	0,43895827		
11912	20070326	26.3.2007	1	0,61865487		
8140	20070327	27.3.2007	2	0,42275442		
7889	20070328	28.3.2007	3	0,40971862		
7350	20070329	29.3.2007	4	0,38172543		
9654	20070330	30.3.2007	5	0,50138466		
8340	20070331	31.3.2007	6	0,4331415		
11238	20070402	2.4.2007	1	0,58365039		
8172	20070403	3.4.2007	2	0,42441635		
9407	20070404	4.4.2007	3	0,48855661		
8042	20070405	5.4.2007	4	0,41766474		
10716		6.4.2007	5	0,55654009		
8561	20070407	7.4.2007	6	0,44461923		
11545		9.4.2007	1	0,59959456		
8701	20070410	10.4.2007	2	0,45189019		

8740	20070411	11.4.2007	3	0,45391568		
8735	20070412	12.4.2007	4	0,453656		
9979	20070413	13.4.2007	5	0,51826368		
8783	20070414	14.4.2007	6	0,4561489		
11773	20070416	16.4.2007	1	0,61143584		
8042	20070417	17.4.2007	2	0,41766474		
7407	20070418	18.4.2007	3	0,38468575		
7979	20070419	19.4.2007	4	0,41439281		
10081	20070420	20.4.2007	5	0,52356109		
8719	20070421	21.4.2007	6	0,45282503		
11335	20070423	23.4.2007	1	0,58868812		
7570	20070424	24.4.2007	2	0,39315122		
7829	20070425	25.4.2007	3	0,4066025		
7403	20070426	26.4.2007	4	0,384478		
10020	20070427	27.4.2007	5	0,52039303		
9234	20070428	28.4.2007	6	0,47957178		
12758	20070430	30.4.2007	1	0,66259224		
7937		1.5.2007	2	0,41221152		
10698	20070502	2.5.2007	3	0,55560525		
8230	20070503	3.5.2007	4	0,42742861		
10323	20070504	4.5.2007	5	0,53612947		
8644	20070505	5.5.2007	6	0,44892987		
12033	20070507	7.5.2007	1	0,62493905		
8304	20070508	8.5.2007	2	0,43127183		
8264	20070509	9.5.2007	3	0,42919441		
8569	20070510	10.5.2007	4	0,44503472		
10876	20070511	11.5.2007	5	0,56484976		
8642	20070512	12.5.2007	6	0,448826		
11676	20070514	14.5.2007	1	0,6063981		
8008	20070515	15.5.2007	2	0,41589894		
8210	20070516	16.5.2007	3	0,4263899		
8504		17.5.2007	4	0,44165891		
12183	20070518	18.5.2007	5	0,63272937		
10122	20070519	19.5.2007	6	0,52569044		
11483	20070521	21.5.2007	1	0,59637457		
8344	20070522	22.5.2007	2	0,43334925		
8157	20070523	23.5.2007	3	0,42363732		
8439	20070524	24.5.2007	4	0,43828311		
10840	20070525	25.5.2007	5	0,56298008		
8865	20070526	26.5.2007	6	0,46040761		
12215	20070528	28.5.2007	1	0,6343913		
7819	20070529	29.5.2007	2	0,40608314		
8356	20070530	30.5.2007	3	0,43397247		
7314	20070531	31.5.2007	4	0,37985575		

9957	20070601	1.6.2007	5	0,5171211		
8755	20070602	2.6.2007	6	0,45469471		
11071	20070604	4.6.2007	1	0,57497717		
8326	20070605	5.6.2007	2	0,43241441		
8338	20070606	6.6.2007	3	0,43303763		
8657	20070607	7.6.2007	4	0,44960504		
10520	20070608	8.6.2007	5	0,54636075		
8793	20070609	9.6.2007	6	0,45666825		
11397	20070611	11.6.2007	1	0,59190812		
8121	20070612	12.6.2007	2	0,42176764		
8138	20070613	13.6.2007	3	0,42265055		
8342	20070614	14.6.2007	4	0,43324537		
10679	20070615	15.6.2007	5	0,55461848		
9080	20070616	16.6.2007	6	0,47157372		
11594	20070618	18.6.2007	1	0,6021394		
8724	20070619	19.6.2007	2	0,45308471		
10573	20070620	20.6.2007	3	0,54911332		
8607	20070621	21.6.2007	4	0,44700826		
10600		22.6.2007	5	0,55051558		
9447		23.6.2007	6	0,49063403		
12402	20070625	25.6.2007	1	0,64410323		
8940	20070626	26.6.2007	2	0,46430276		
8668	20070627	27.6.2007	3	0,45017633		
8872	20070628	28.6.2007	4	0,46077115		
10521	20070629	29.6.2007	5	0,54641268		
9815	20070630	30.6.2007	6	0,50974627		
13990	20070702	2.7.2007	1	0,72657669		
10688	20070703	3.7.2007	2	0,5550859		
10096	20070704	4.7.2007	3	0,52434012		
9801	20070705	5.7.2007	4	0,50901917		
12030	20070706	6.7.2007	5	0,62478325		
10388	20070707	7.7.2007	6	0,53950527		
13159	20070709	9.7.2007	1	0,68341835		
9612	20070710	10.7.2007	2	0,49920337		
9366	20070711	11.7.2007	3	0,48642726		
9121	20070712	12.7.2007	4	0,47370308		
11904	20070713	13.7.2007	5	0,61823938		
10372	20070714	14.7.2007	6	0,5386743		
12925	20070716	16.7.2007	1	0,67126546		
10036	20070717	17.7.2007	2	0,521224		
9778	20070718	18.7.2007	3	0,50782465		
9216	20070719	19.7.2007	4	0,47863694		
12030	20070720	20.7.2007	5	0,62478325		
10150	20070721	21.7.2007	6	0,52714464		

12741	20070723	23.7.2007	1	0,66170934		
9840	20070724	24.7.2007	2	0,51104465		
9149	20070725	25.7.2007	3	0,47515727		
9147	20070726	26.7.2007	4	0,4750534		
11668	20070727	27.7.2007	5	0,60598262		
10296	20070728	28.7.2007	6	0,53472721		
12814	20070730	30.7.2007	1	0,66550063		
10143	20070731	31.7.2007	2	0,52678109		
8359	20070801	1.8.2007	3	0,43412828		
8370	20070802	2.8.2007	4	0,43469957		
10620	20070803	3.8.2007	5	0,55155429		
9368	20070804	4.8.2007	6	0,48653113		
11696	20070806	6.8.2007	1	0,60743681		
8496	20070807	7.8.2007	2	0,44124343		
8237	20070808	8.8.2007	3	0,42779215		
8075	20070809	9.8.2007	4	0,41937861		
10394	20070810	10.8.2007	5	0,53981688		
9252	20070811	11.8.2007	6	0,48050662		
11928	20070813	13.8.2007	1	0,61948583		
8148	20070814	14.8.2007	2	0,4231699		
8462	20070815	15.8.2007	3	0,43947763		
8290	20070816	16.8.2007	4	0,43054473		
10344	20070817	17.8.2007	5	0,53722011		
9032	20070818	18.8.2007	6	0,46908082		
11749	20070820	20.8.2007	1	0,61018939		
8376	20070821	21.8.2007	2	0,43501118		
8178	20070822	22.8.2007	3	0,42472796		
7900	20070823	23.8.2007	4	0,41028991		
10172	20070824	24.8.2007	5	0,52828721		
8770	20070825	25.8.2007	6	0,45547374		
11537	20070827	27.8.2007	1	0,59917908		
8153	20070828	28.8.2007	2	0,42342958		
7990	20070829	29.8.2007	3	0,4149641		
8164	20070830	30.8.2007	4	0,42400087		
10900	20070831	31.8.2007	5	0,56609621		
10220	20070901	1.9.2007	6	0,53078012		
12286	20070903	3.9.2007	1	0,63807872		
8313	20070904	4.9.2007	2	0,43173925		
8549	20070905	5.9.2007	3	0,44399601		
9057	20070906	6.9.2007	4	0,47037921		
12104	20070907	7.9.2007	5	0,62862647		
11180	20070908	8.9.2007	6	0,58063813		
12398	20070910	10.9.2007	1	0,64389549		
9268	20070911	11.9.2007	2	0,48133758		

9129	20070912	12.9.2007	3	0,47411856		
8373	20070913	13.9.2007	4	0,43485537		
12460	20070914	14.9.2007	5	0,64711548		
11084	20070915	15.9.2007	6	0,57565233		
12537	20070917	17.9.2007	1	0,65111451		
8624	20070918	18.9.2007	2	0,44789117		
8296	20070919	19.9.2007	3	0,43085634		
7988	20070920	20.9.2007	4	0,41486023		
10212	20070921	21.9.2007	5	0,53036463		
9874	20070922	22.9.2007	6	0,51281046		
12182	20070924	24.9.2007	1	0,63267743		
7923	20070925	25.9.2007	2	0,41148443		
8220	20070926	26.9.2007	3	0,42690925		
8117	20070927	27.9.2007	4	0,4215599		
10771	20070928	28.9.2007	5	0,55939654		
9984	20070929	29.9.2007	6	0,51852335		

LIITE 4

Koe-ennustukset

6 X 3 X 1 neuroverkko

Kierrokset													
10	20	30	40	50	60	70	80	90	100	110	120	130	200
OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1
0,3295746	0,4278464	0,4926932	0,4939594	0,4894168	0,4882674	0,4882213	0,4881869	0,4878879	0,4872466	0,4862162	0,4847114	0,4825607	0,4541054
0,496172	0,5587597	0,6189835	0,6370199	0,6366738	0,6339619	0,6309595	0,6276171	0,6238709	0,6196357	0,6148138	0,6092641	0,6027478	0,5504336
0,4850748	0,4400151	0,4575933	0,4740155	0,4731988	0,4684975	0,463056	0,4567981	0,4496261	0,4414429	0,4321163	0,4214541	0,409142	0,3198987
0,6580807	0,5600471	0,4861506	0,4758345	0,4713123	0,4661014	0,4609155	0,4552436	0,4487749	0,441376	0,4329356	0,4233012	0,4122189	0,3319132
0,5583733	0,5157199	0,4914085	0,4833489	0,4800696	0,4757067	0,4704193	0,4643639	0,457462	0,4496363	0,4408028	0,430837	0,4195301	0,3446669
0,5430575	0,6004882	0,6270315	0,6241761	0,6186417	0,6131994	0,6069189	0,5996727	0,5914336	0,5821384	0,5716935	0,5599621	0,5467306	0,4624508
0,3972386	0,5332238	0,6112001	0,612241	0,6016715	0,592232	0,5826241	0,5719273	0,5598001	0,5460149	0,5303406	0,5124773	0,4919952	0,358582
0,577431	0,6339679	0,6853423	0,6992379	0,6943122	0,6858861	0,6762653	0,6653873	0,6530819	0,6391185	0,6232196	0,6050269	0,5840312	0,4421372
0,5800447	0,5220293	0,5161055	0,5229531	0,5158295	0,5040111	0,4905269	0,4752539	0,4579501	0,4383666	0,4162056	0,3910892	0,3624909	0,1802288
0,700326	0,6269634	0,5644867	0,5494256	0,5381058	0,5249277	0,5104275	0,4941393	0,4756753	0,4547492	0,4310587	0,4042348	0,3737828	0,183809
0,6210016	0,6098301	0,5922939	0,5771294	0,5655545	0,5529467	0,5386778	0,5225407	0,5042392	0,4834963	0,460014	0,4334285	0,4032609	0,2194685
0,6103913	0,676307	0,701313	0,6967079	0,6889679	0,6803713	0,6703506	0,6588746	0,6458263	0,6310186	0,6142239	0,5951529	0,5734131	0,4370918
0,423977	0,5783957	0,6631868	0,665827	0,6551853	0,6440758	0,6317862	0,617779	0,6017914	0,5835466	0,5627214	0,5389057	0,5115461	0,3371089
0,5908346	0,6602444	0,7063945	0,7138002	0,7059459	0,6958254	0,684641	0,67201	0,6576389	0,6412254	0,622424	0,6007971	0,5757405	0,4074532
0,5754747	0,5536257	0,5723153	0,5821177	0,5744657	0,5616127	0,5467469	0,5298634	0,5107073	0,4889646	0,4642607	0,4361278	0,4039328	0,1986688
0,7180538	0,6457251	0,5844877	0,5729297	0,5639178	0,5522728	0,5391379	0,5243306	0,5075614	0,4885818	0,4671173	0,4428286	0,4152534	0,2416029
0,6036677	0,5783521	0,5472111	0,5275141	0,5173606	0,5073693	0,4962103	0,483679	0,4695173	0,4535288	0,4355196	0,415246	0,3923681	0,2532218
0,5778912	0,6649446	0,7006928	0,6982525	0,6924602	0,6861494	0,6786148	0,6699444	0,6601228	0,6490452	0,6365714	0,6225145	0,6066079	0,5076707
0,3907881	0,549486	0,6317364	0,6338331	0,6270844	0,6216957	0,6162414	0,6101858	0,6033491	0,5956061	0,5868366	0,5768742	0,5654593	0,489205
0,4887548	0,5884195	0,6622387	0,6798098	0,6793008	0,676442	0,673034	0,6691849	0,6648908	0,6600787	0,6546544	0,648482	0,641335	0,5878821
0,4517015	0,4405523	0,477382	0,4949105	0,4943699	0,4907275	0,4865812	0,4817874	0,47626	0,4699169	0,4626463	0,4542804	0,444536	0,3705465
0,6386464	0,5477683	0,4845035	0,4790118	0,4772838	0,4748822	0,4728019	0,4706067	0,4680203	0,4649364	0,4612781	0,4569339	0,4516979	0,403786
0,5332951	0,4775767	0,4528025	0,4501374	0,4516497	0,4516448	0,4508932	0,449784	0,448338	0,446542	0,4443725	0,4417725	0,4386115	0,4082687
0,5214087	0,5590791	0,5733198	0,569364	0,5671025	0,566053	0,5647722	0,5631055	0,561066	0,5586649	0,5558981	0,5527326	0,5490769	0,5191315
0,3443438	0,4822952	0,5633068	0,5660422	0,5600524	0,5563898	0,5532897	0,5499147	0,5460218	0,5415017	0,5362672	0,5301891	0,5230458	0,4688919

Koe-ennustukset

6 X 12 X 1 neuroverkko

Kierrokset							
10	20	30	40	50	60	70	80
OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1	OUT1
0,502961	0,4892493	0,4926932	0,4737211	0,4627426	0,4551031	0,4497867	0,4464413
0,541072	0,5791262	0,6189835	0,597995	0,606299	0,6086112	0,6065421	0,600917
0,3572934	0,3271513	0,4575933	0,3462992	0,366825	0,3801669	0,3868119	0,3881936
0,3566554	0,3315483	0,4861506	0,3435672	0,3552113	0,3616022	0,3637357	0,3629668
0,4565088	0,4113888	0,4914085	0,3838405	0,3686672	0,3598124	0,3542095	0,3499579
0,5278309	0,5298547	0,6270315	0,5170384	0,5043915	0,4944015	0,4865619	0,4800096
0,522604	0,5102185	0,6112001	0,493626	0,4793851	0,4662947	0,4530947	0,4390529
0,5222154	0,5546935	0,6853423	0,5757461	0,5849632	0,5857534	0,5796907	0,567458
0,3263736	0,2762339	0,5161055	0,289858	0,3074263	0,3171483	0,3187539	0,3132477
0,3400253	0,2929887	0,5644867	0,2942702	0,2990456	0,2992872	0,2947931	0,2860899
0,4619023	0,4083662	0,5922939	0,3720273	0,348084	0,330155	0,3147184	0,2993566
0,5741194	0,5816489	0,701313	0,5636314	0,5442445	0,527429	0,5123153	0,4973602
0,5428351	0,531097	0,6631868	0,5108212	0,4931609	0,477366	0,4616989	0,4447582
0,5275052	0,5559414	0,7063945	0,5708736	0,5747606	0,5710813	0,56114	0,5454263
0,3850103	0,348016	0,5723153	0,3574586	0,3690722	0,373466	0,3699549	0,3588599
0,3627869	0,3322101	0,5844877	0,339297	0,3462862	0,3478287	0,3443098	0,3363484
0,4662276	0,4071813	0,5472111	0,3632578	0,3347446	0,3152875	0,3008591	0,2886681
0,6016165	0,6179536	0,7006928	0,600565	0,5809251	0,5646559	0,551157	0,5389108
0,5949895	0,6023483	0,6317364	0,5826749	0,5631738	0,5471297	0,5334851	0,5210098
0,5888455	0,6326914	0,6622387	0,645667	0,6485503	0,6472334	0,6428453	0,6356232
0,3975653	0,3794652	0,477382	0,3957158	0,4119497	0,4221954	0,4269475	0,4273976
0,3930862	0,384121	0,4845035	0,3988173	0,4105025	0,4168795	0,4192752	0,4190712
0,4608448	0,4239827	0,4528025	0,40211	0,3914311	0,3867222	0,3852925	0,3855012
0,5250504	0,5294366	0,5733198	0,5172932	0,5057827	0,4980122	0,493686	0,4921153
0,552827	0,5501735	0,5633068	0,5322977	0,5167316	0,5044717	0,494399	0,485699

LIITE 5

Valitulla neuroverkolla tuotetut ennusteet

MP	INPUT1	INPUT2	INPUT3	INPUT4	INPUT5	INPUT6	OUT1
1	0,455473739	0,59917908	0,423429578	0,414964102	0,424000867	0,566096209	0,486216241
2	0,59917908	0,423429578	0,414964102	0,424000867	0,566096209	0,530780116	0,614813837
3	0,423429578	0,414964102	0,424000867	0,566096209	0,530780116	0,638078718	0,432116299
4	0,414964102	0,424000867	0,566096209	0,530780116	0,638078718	0,431739247	0,432935571
5	0,424000867	0,566096209	0,530780116	0,638078718	0,431739247	0,443996009	0,440802792
6	0,566096209	0,530780116	0,638078718	0,431739247	0,443996009	0,470379208	0,571693454
7	0,530780116	0,638078718	0,431739247	0,443996009	0,470379208	0,62862647	0,530340578
8	0,638078718	0,431739247	0,443996009	0,470379208	0,62862647	0,58063813	0,623219601
9	0,431739247	0,443996009	0,470379208	0,62862647	0,58063813	0,643895487	0,416205627
10	0,443996009	0,470379208	0,62862647	0,58063813	0,643895487	0,481337584	0,43105866
11	0,470379208	0,62862647	0,58063813	0,643895487	0,481337584	0,474118559	0,460013964
12	0,62862647	0,58063813	0,643895487	0,481337584	0,474118559	0,434855373	0,614223886
13	0,58063813	0,643895487	0,481337584	0,474118559	0,434855373	0,647115483	0,562721447
14	0,643895487	0,481337584	0,474118559	0,434855373	0,647115483	0,575652329	0,622424028
15	0,481337584	0,474118559	0,434855373	0,647115483	0,575652329	0,651114512	0,464260668
16	0,474118559	0,434855373	0,647115483	0,575652329	0,651114512	0,447891166	0,467117335
17	0,434855373	0,647115483	0,575652329	0,651114512	0,447891166	0,430856344	0,435519648
18	0,647115483	0,575652329	0,651114512	0,447891166	0,430856344	0,414860231	0,636571367
19	0,575652329	0,651114512	0,447891166	0,430856344	0,414860231	0,530364632	0,586836587
20	0,651114512	0,447891166	0,430856344	0,414860231	0,530364632	0,512810456	0,654654447
21	0,447891166	0,430856344	0,414860231	0,530364632	0,512810456	0,632677433	0,462646308
22	0,430856344	0,414860231	0,530364632	0,512810456	0,632677433	0,411484428	0,461278147
23	0,414860231	0,530364632	0,512810456	0,632677433	0,411484428	0,426909251	0,44437254
24	0,530364632	0,512810456	0,632677433	0,411484428	0,426909251	0,421559902	0,555898143
25	0,512810456	0,632677433	0,411484428	0,426909251	0,421559902	0,559396539	0,536267237